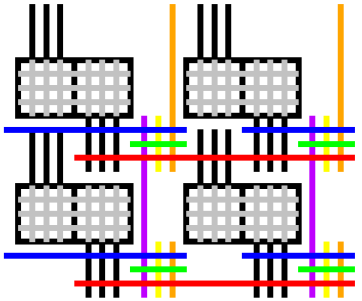
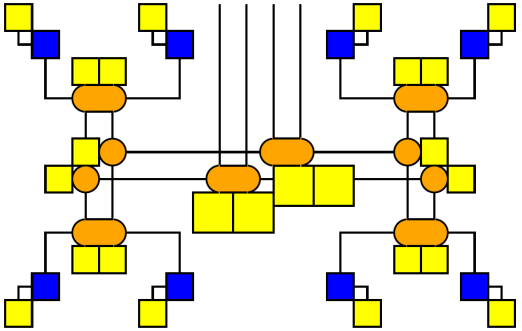




Understanding and Exploiting the Energy advantages of Field-Programmable Technologies

André DeHon

andre@seas.upenn.edu

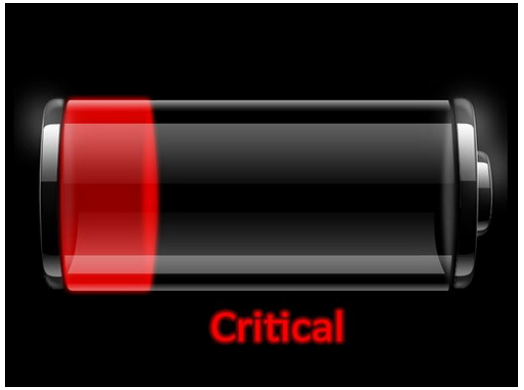
Penn

Outline

- Energy: why care?
- Teasers
- Architecture
- Implication for
  - Applications
  - FPGA Architecture
- Information (Coding)
- Variation and Aging

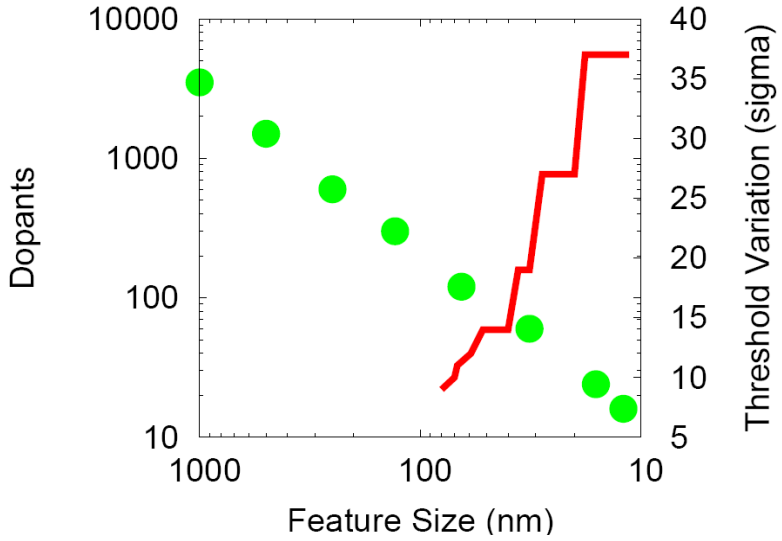
Energy Setup

Energy



- Growing domain of portables
  - Less energy/op → longer battery life
- Global Energy Crisis
- Power-envelope at key limit
  - E reduce → increase compute in P-envelope
  - Scaling
    - Power density **not** transistors limit sustained ops/s
  - Server rooms
    - Cost-of-ownership **not** dominated by Silicon
      - Cooling, Power bill

Energy, Reliability, Capacity Squeeze

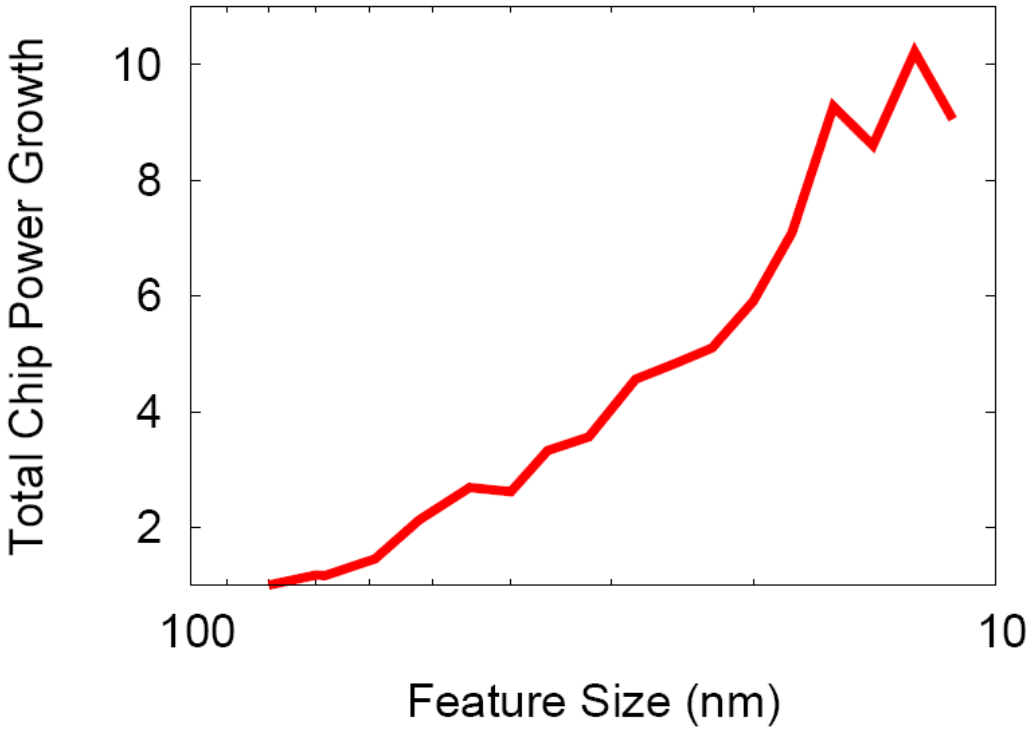


- Smaller devices→higher variation→increased voltage margins
  - ITRS shows mostly flat V_{dd}
- Capacitance/gate decreases little
  - Maybe halved 45nm to 11nm [ITRS]
  - When Gate count increase 16× ?!?
- Power density already limiting use of transistor capacity on die

$$P \propto N_{gates} C_g V^2 f$$

Trends threaten ability to exploit smaller technologies.

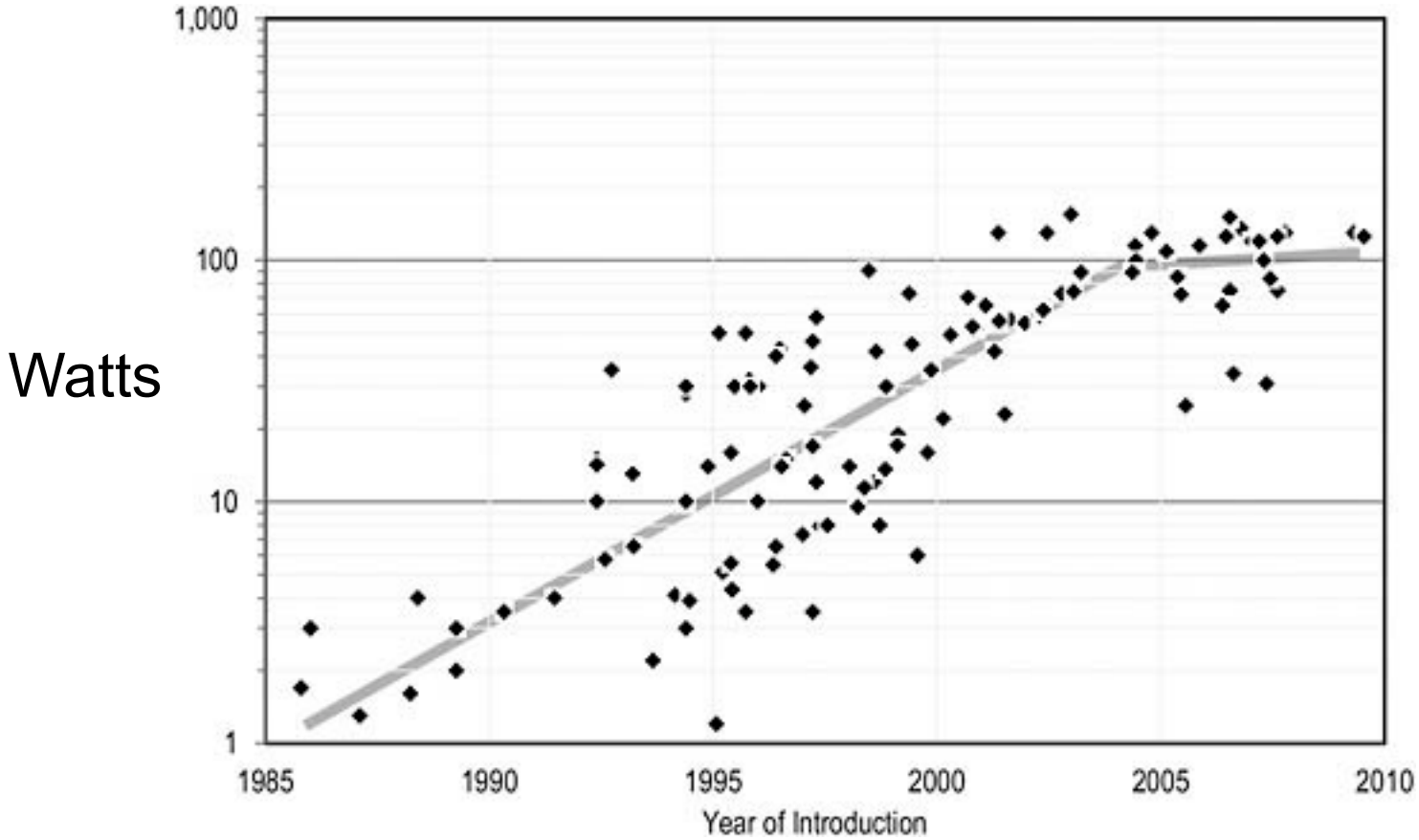
Power Gap



- Based on ITRS2009 data
- Assume start power-limited designs, gap grows with continued scaling

$$P \propto N_{gates} C_g V^2 f$$

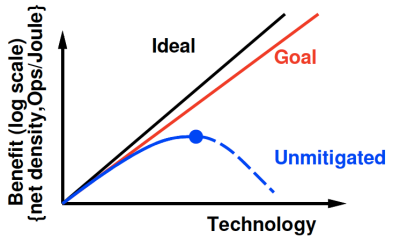
uP Power Density

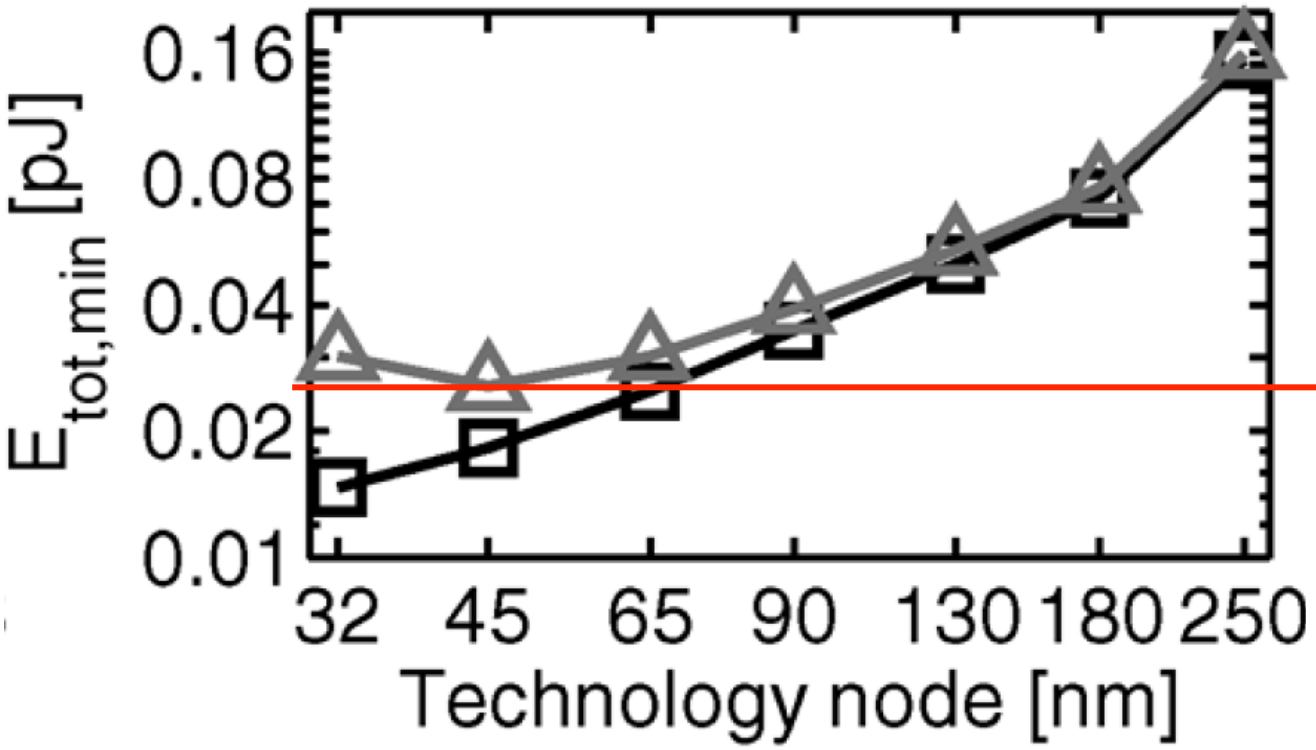


The Future of Computing Performance: Game Over or Next Level?
National Academy Press, 2011

http://www.nap.edu/catalog.php?record_id=12980

Variation threatens E/Op reduction





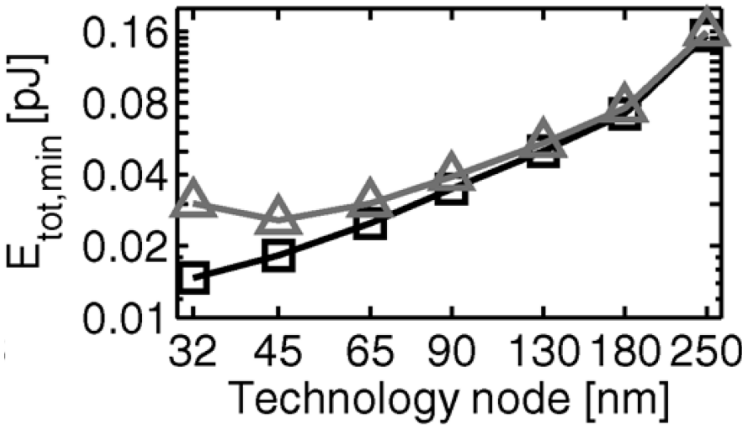
Black nominal

Grey with variation

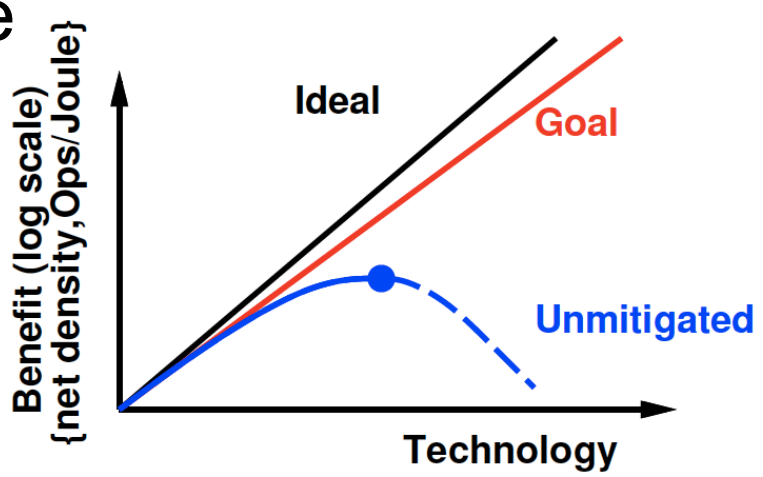
Min-Energy for multiplication (typically subthreshold)

[Bol *et al*., IEEE TR VLSI Sys 17(10):1508—1519]

Energy Limited



- It is Energy that defines
  - Ops/s can extract from a power-limited chip
  - Ops/battery-hour can extract from a portable
- If a technology makes E/op worse
  - That technology is worse
  - End-of-scaling



Energy

$$E \propto \alpha C V^2$$

- C – driven by architecture
  - Also impacted by variation, aging
- V – today, driven by variation, aging
- α – driven by architecture, coding/information

Energy

Architecture

Variation/Aging

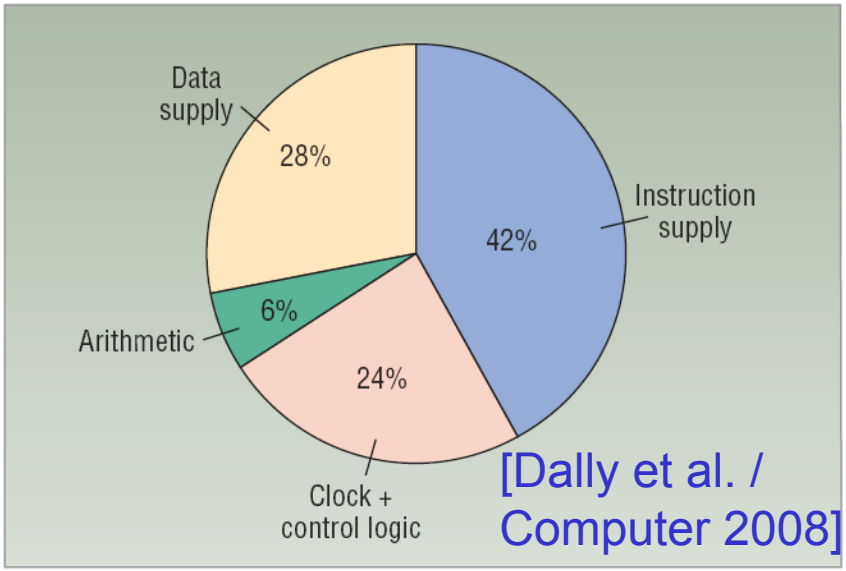
$$E \propto \alpha C V^2$$

Information

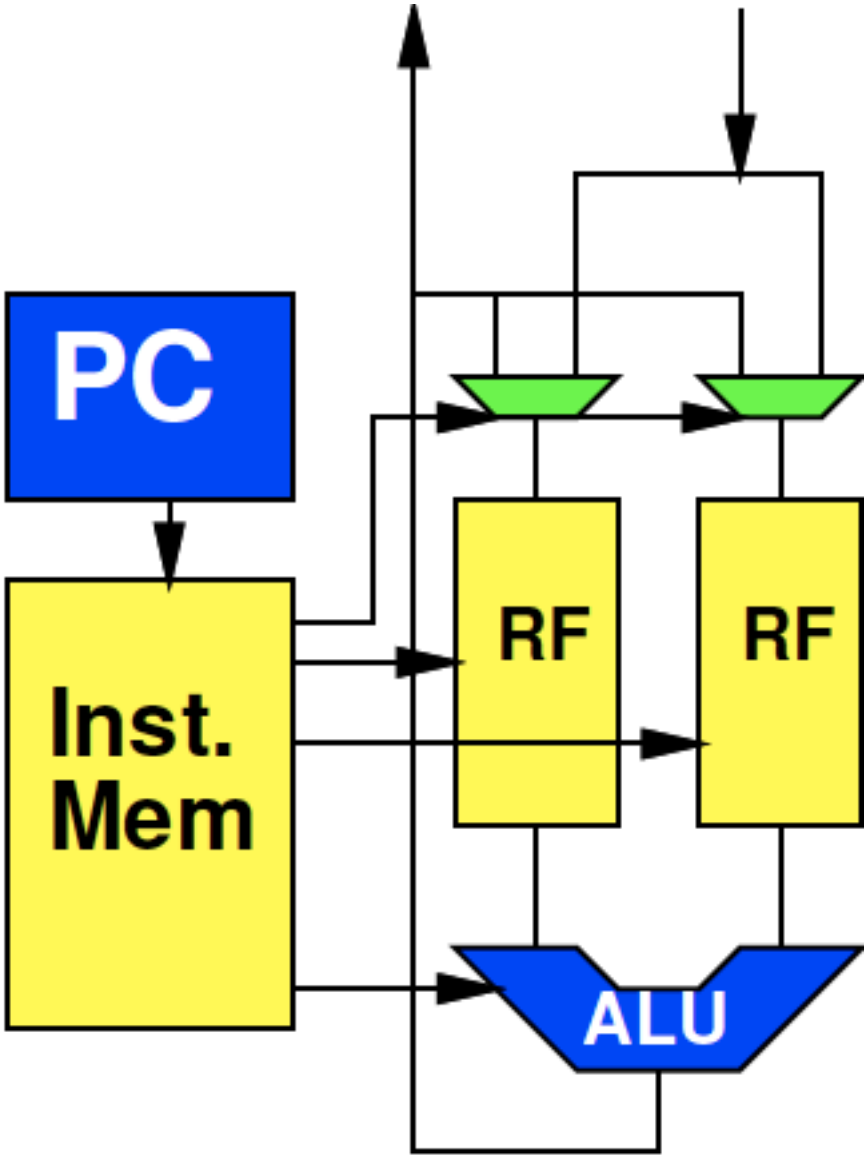
Teasers

Processors and Energy

- Very little into actual computation
- Determine and Fetch Instruction
- Read and Write data from memories

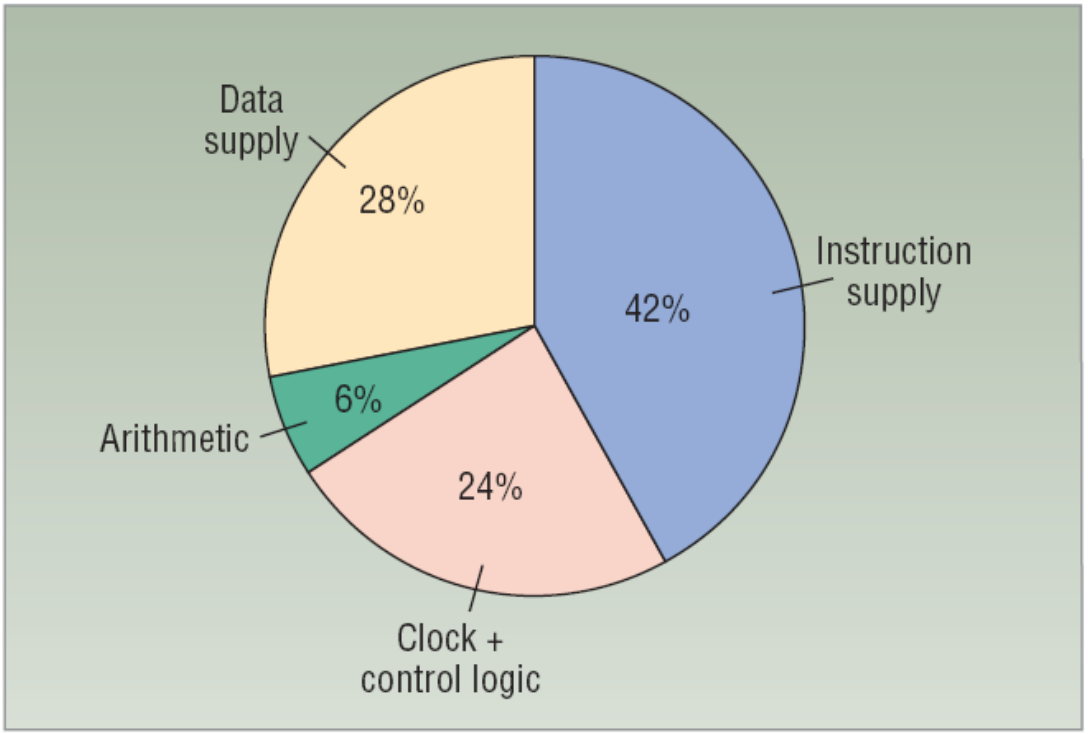


[Dally et al. / Computer 2008]



FPGA Energy Advantage

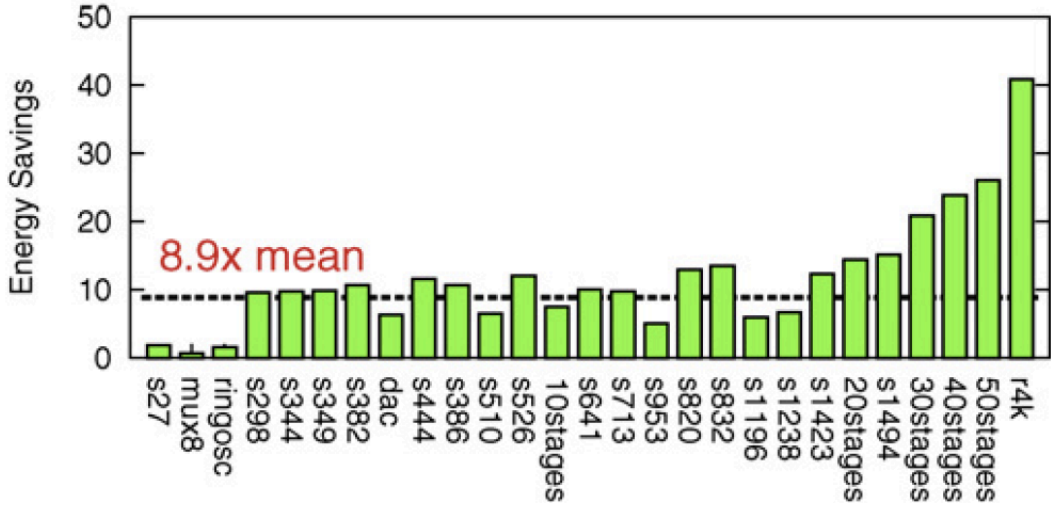
- No memory energy
- Exploit data correlation (Information)
- Performance with low clock rate
- Fine-grained specialization
  - Just the needed operations



[Dally et al. / Computer 2008]

Energy-Efficiency of Spatial Computation

- Considerable evidence of lower energy on FPGAs than Processors and GPUs
- SPICE simulation example
  - V6LX760 (40nm) vs. Core i7 965 (45nm)



[Kapre+DeHon/TRCAD v31n1p9]

More Preliminary Evidence

- Microsoft Catapult justifies FPGAs in Cloud servers by Energy reduction
- FPGAs for finance driven by limited power-density in servers "near" trading floor
- Rush to "accelerators" driven by energy inefficiency of processor
- This conference, FCCM
  - Full of papers showing FPGA energy advantage

Energy Advantage FPGAs

- Why?
- Potential for additional advantage?
- How do we increase advantage?

Outline

- Energy: why care?
- Teasers
- Architecture
- Implication for
  - Applications
  - FPGA Architecture
- Information (Coding)
- Variation and Aging

$$E \propto \alpha C V^2$$

Architecture

E(spatial)<E(sequential)

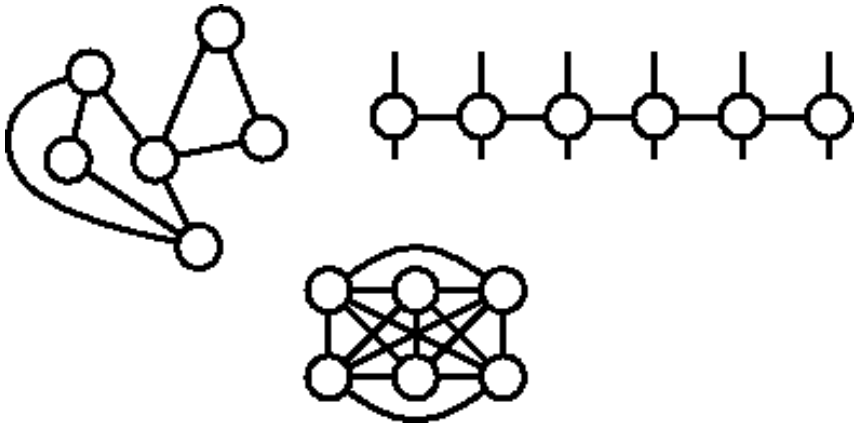
E(FPGA)<E(processor)

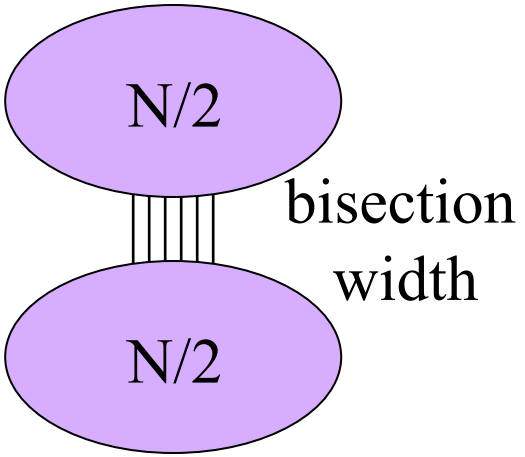
Architecture Outline

- Rent's Rule and VLSI Complexity
- Memories
- Central Processor
- Spatial Locality
  - Data
  - Instruction
- Wire Sharing

Bisection Width

- Partition design into two equal size halves
  - Minimize wires (nets) with ends in both halves
- Number of wires crossing is **bisection width**
  - Information crossing
- lower bw = more locality





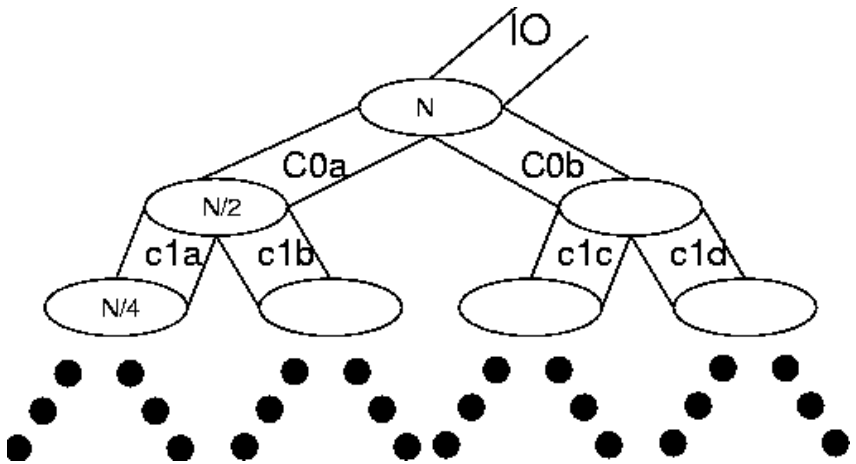
Rent's Rule

- If we recursively bisect a graph, attempting to minimize the cut size, we typically get:

$$BW = IO = c\,N^{p}$$

  - $0 \le p \le 1$
  - $p \le 1$ means many inputs come from within a partition

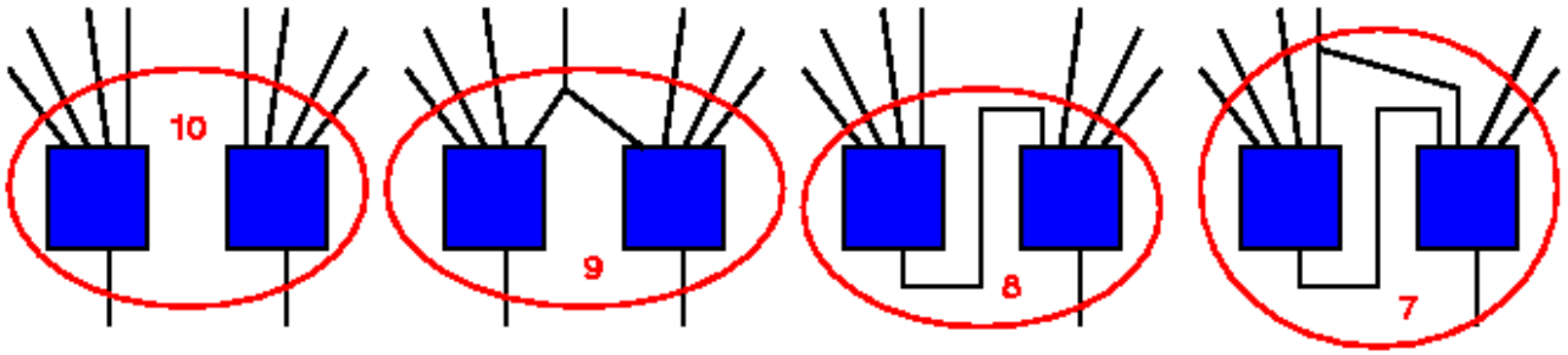
[Landman and Russo, IEEE TR Computers p1469, 1971]



Rent and Locality

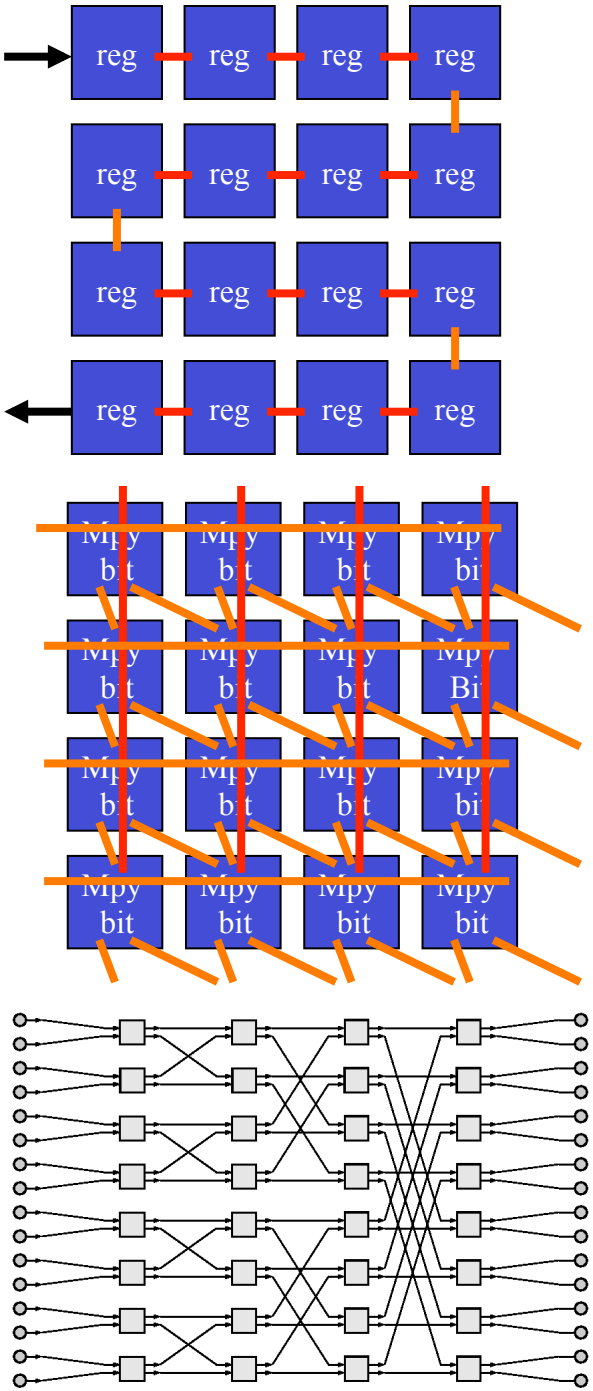
- Rent and IO quantifying locality
  - local consumption
  - local fanout

$$IO = c\ N^{p}$$



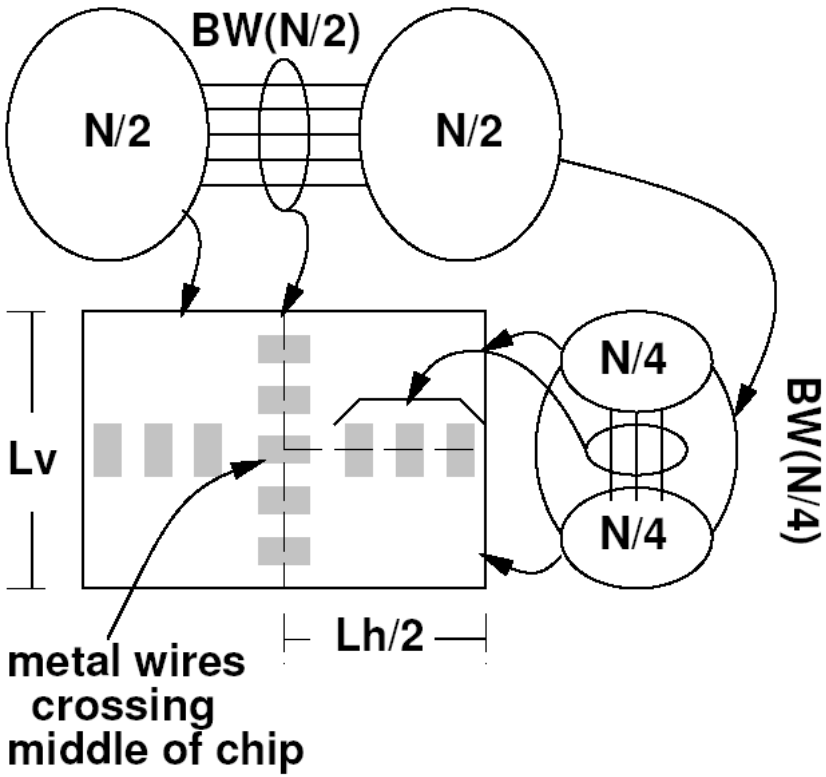
Common Applications

- Rent p=0
  - Shift-register, 1D filter
- Rent p=0.5
  - Array multiplier
  - 2D Window Filter
  - nearest-neighbor
- Rent p=1.0
  - FFT, Sort

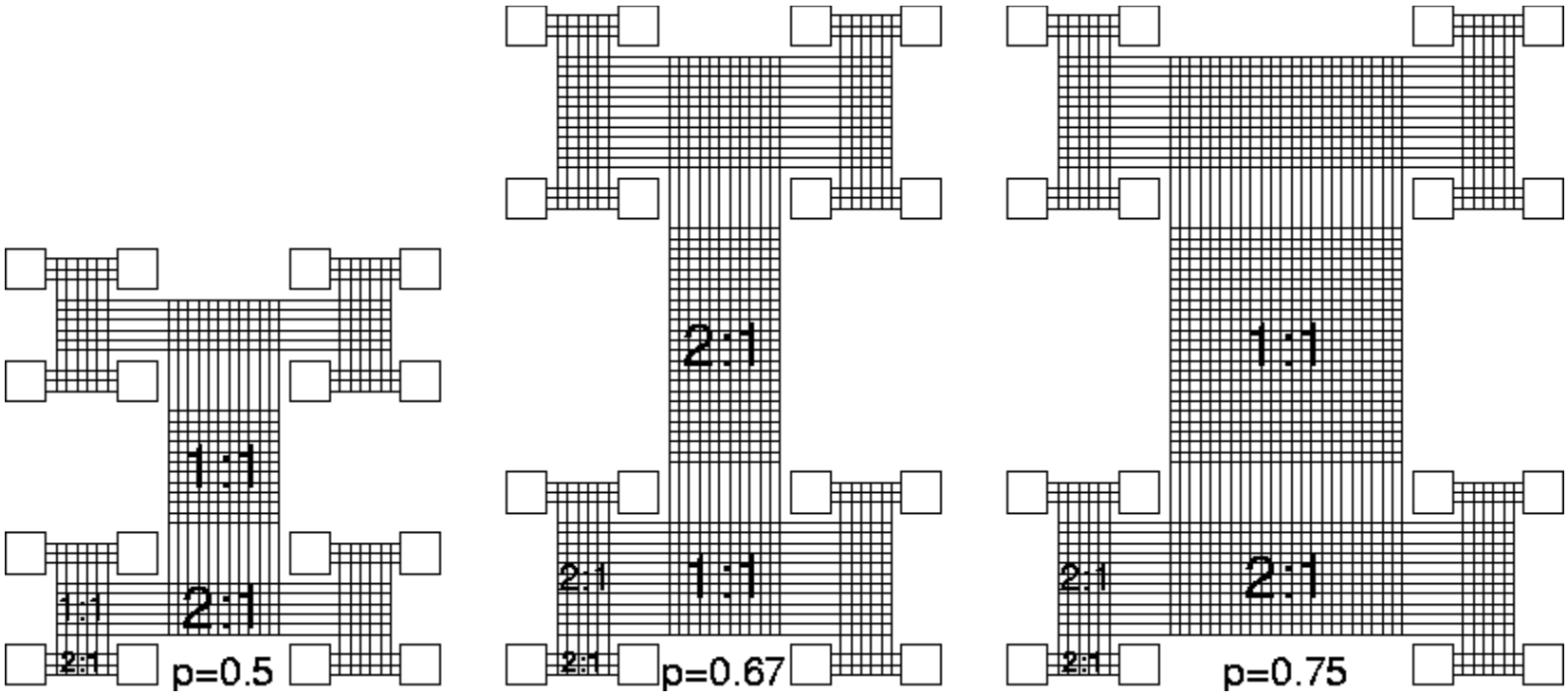


VLSI Interconnect Area

- Bisection width is lower-bound on IC width
  - When wire dominated, may be tight bound
- (recursively)
- Rent's Rule tells us how big our chip must be



Rent Network Richness

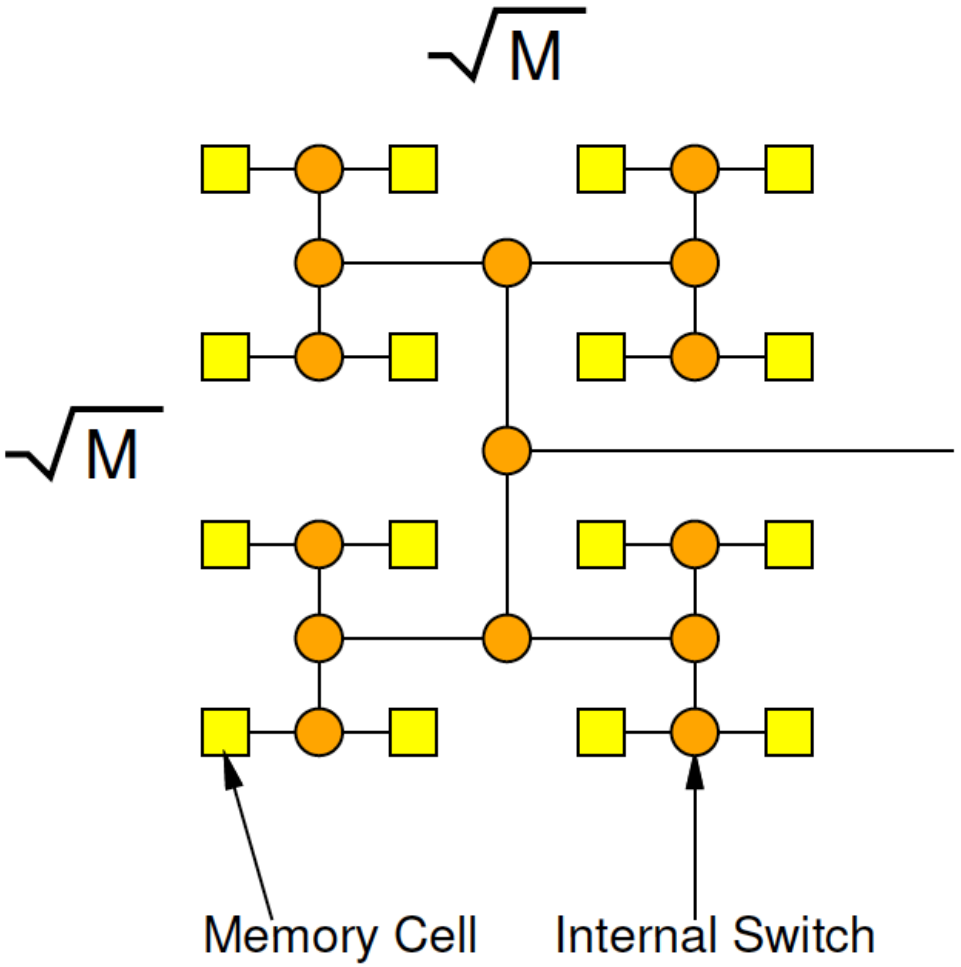


Architecture Outline

- Rent's Rule and VLSI Complexity
- Memories
- Central Processor
- Spatial Locality
  - Data
  - Instruction
- Wire Sharing

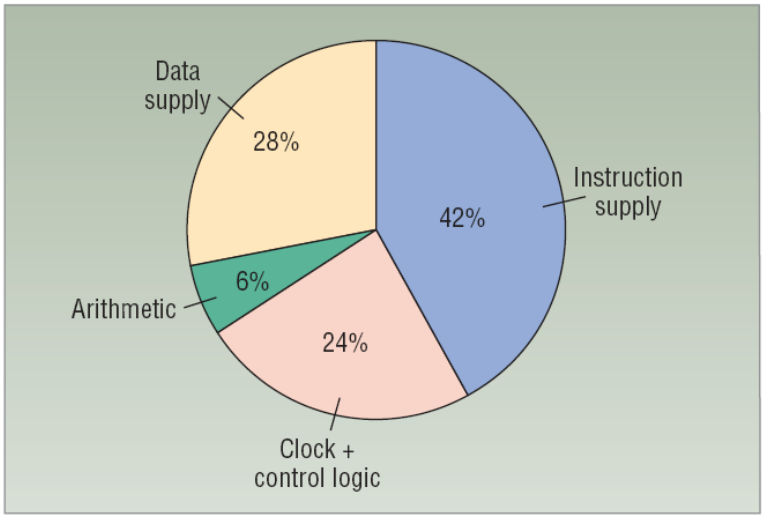
Memory Energy

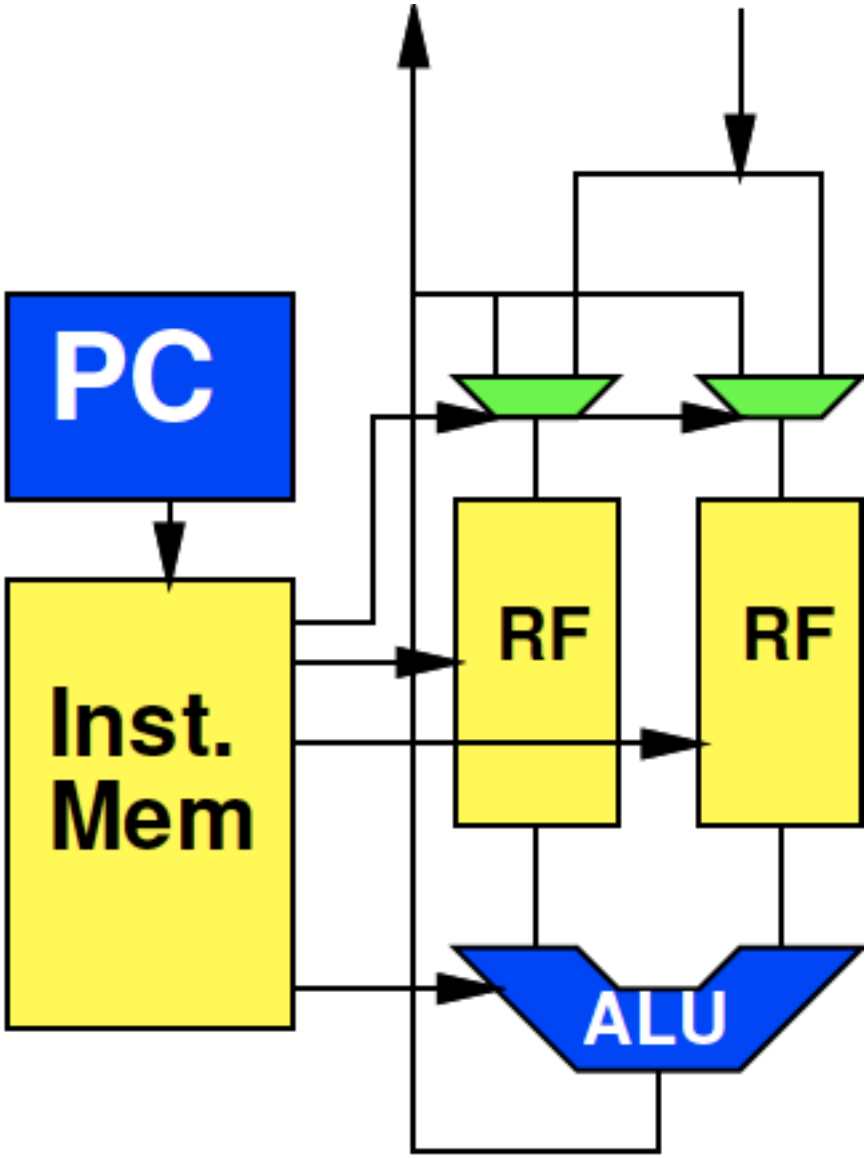
- Reading out of large memories expensive
- The larger the memory, the more expensive per **bit** read
- $O(M^{0.5})$ wire length
- Random access must send address
  - $O(M^{0.5}\log(M))$
- Sequential access
  - $O(M^{0.5})$ per bit



Processors and Energy

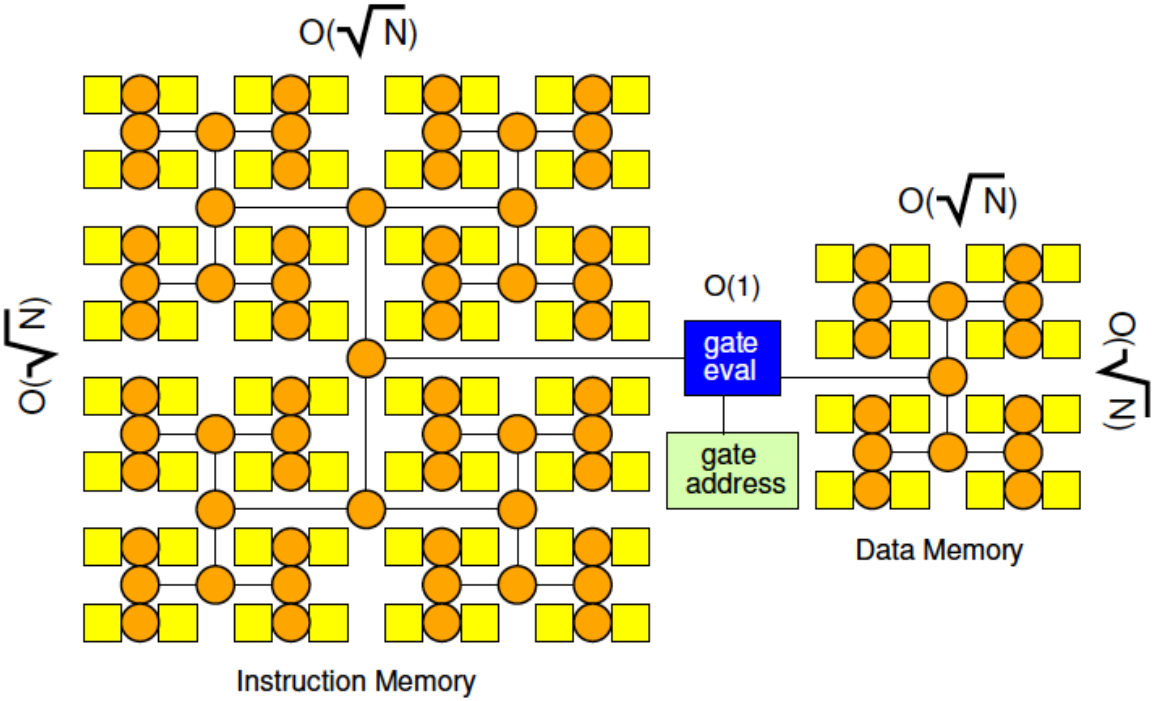
- Most energy reading out of two memories
  - Instruction
  - Data





Central Processor with Description Locality

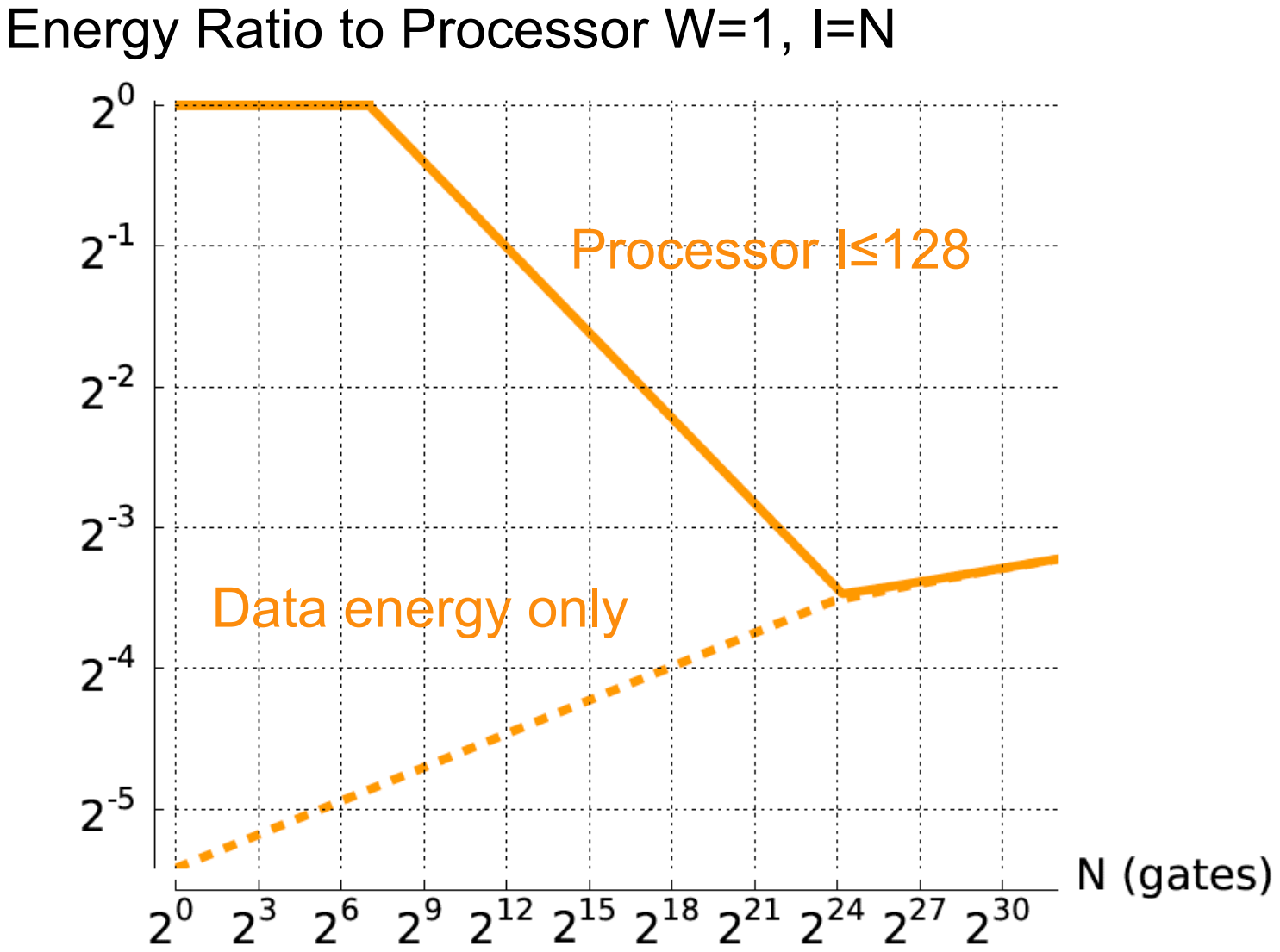
- p<1.0, total instruction bits are O(N)
  - $O(N^{1.5})$
- Read/write O(N) bits
  - $O(N^{1.5}\log(N))$



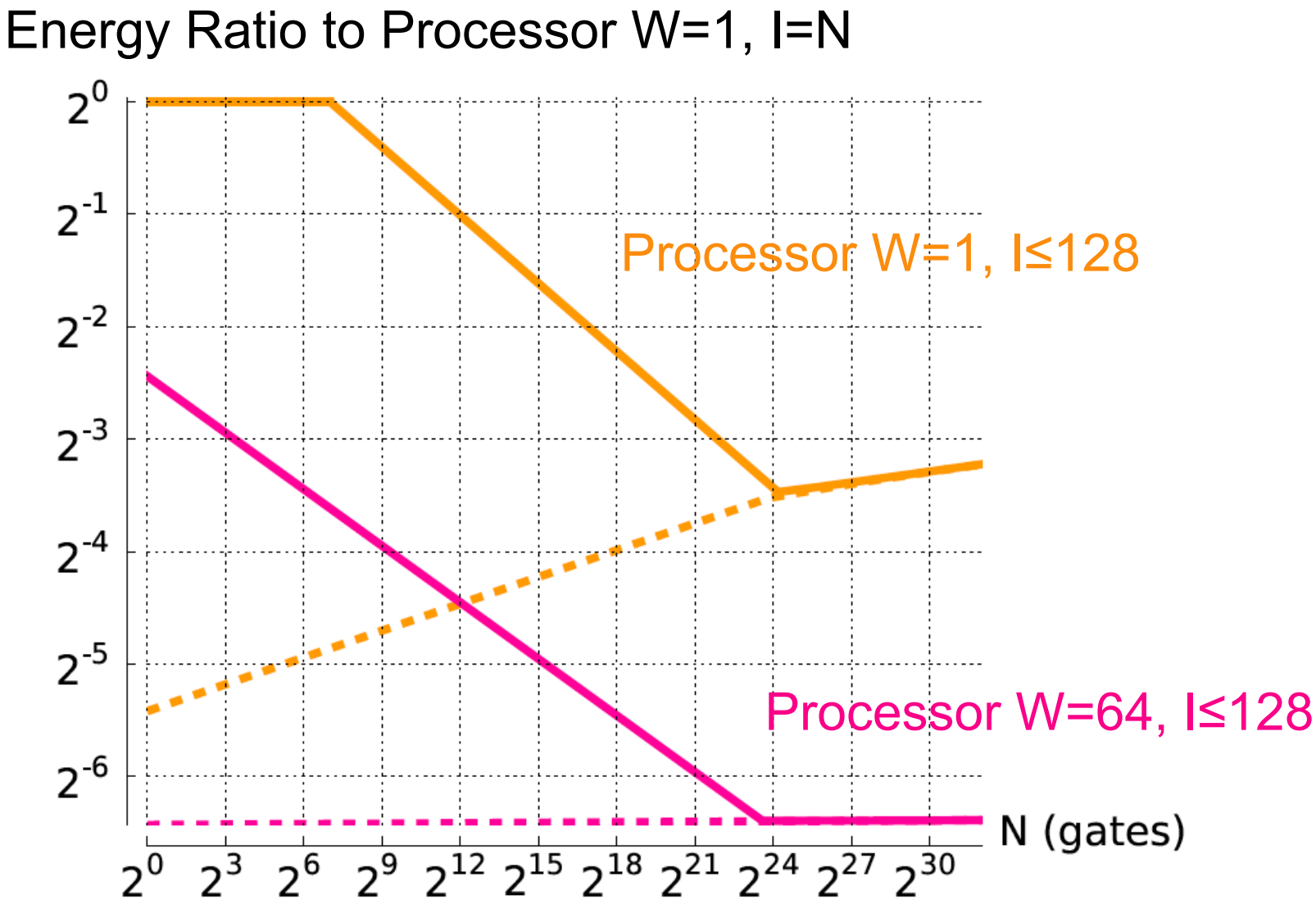
Instruction Sharing

- **Problem:** absolutely, instruction energy dominates data energy
- **Assumption:** every bit operator needs its own unique instruction
- **Opportunity:** share instructions across operators
  - Looping, wide words
- Still $O(N^{1.5})$ data memory energy

Instruction Sharing (I) (p=0.7)



SIMD+Instr. Share (p=0.7)

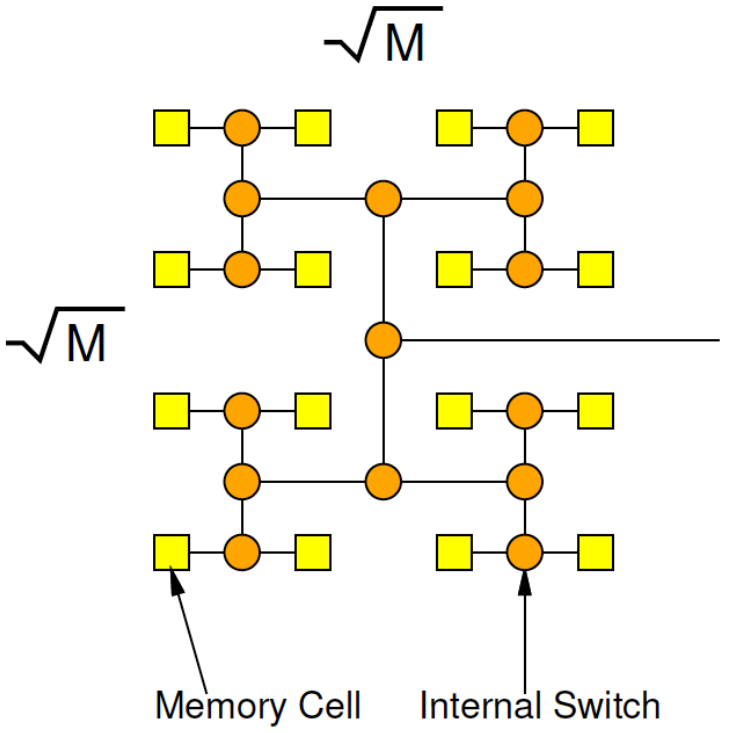


Architecture Outline

- Rent's Rule and VLSI Complexity
- Memories
- Central Processor
- Spatial Locality
  - Data
  - Instruction
- Wire Sharing

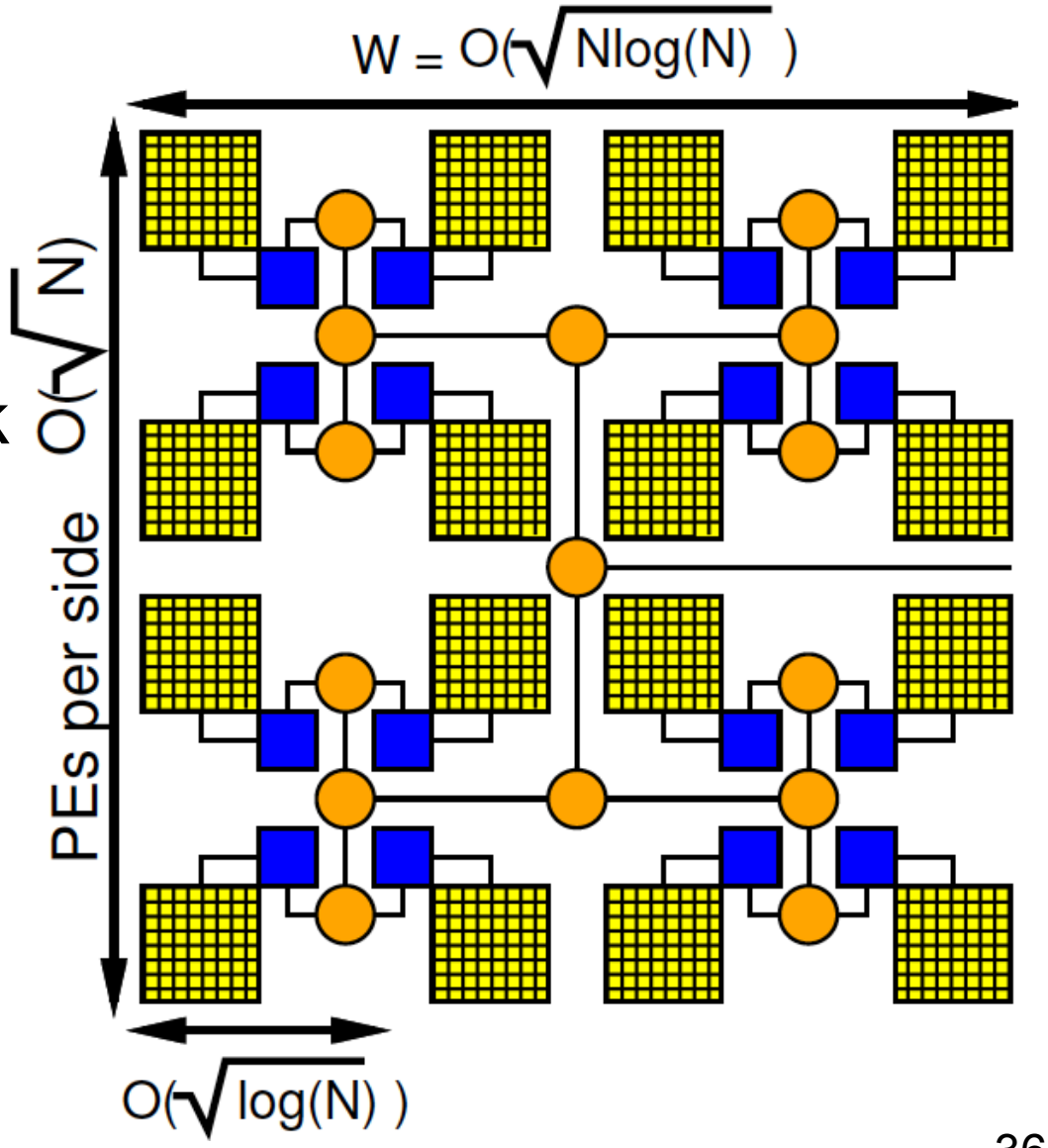
Data Locality

- **Problem:** Must pay $O(N^{0.5})$ for every read since data must be moved in and out of memory.
- **Opportunity:** compute local to data



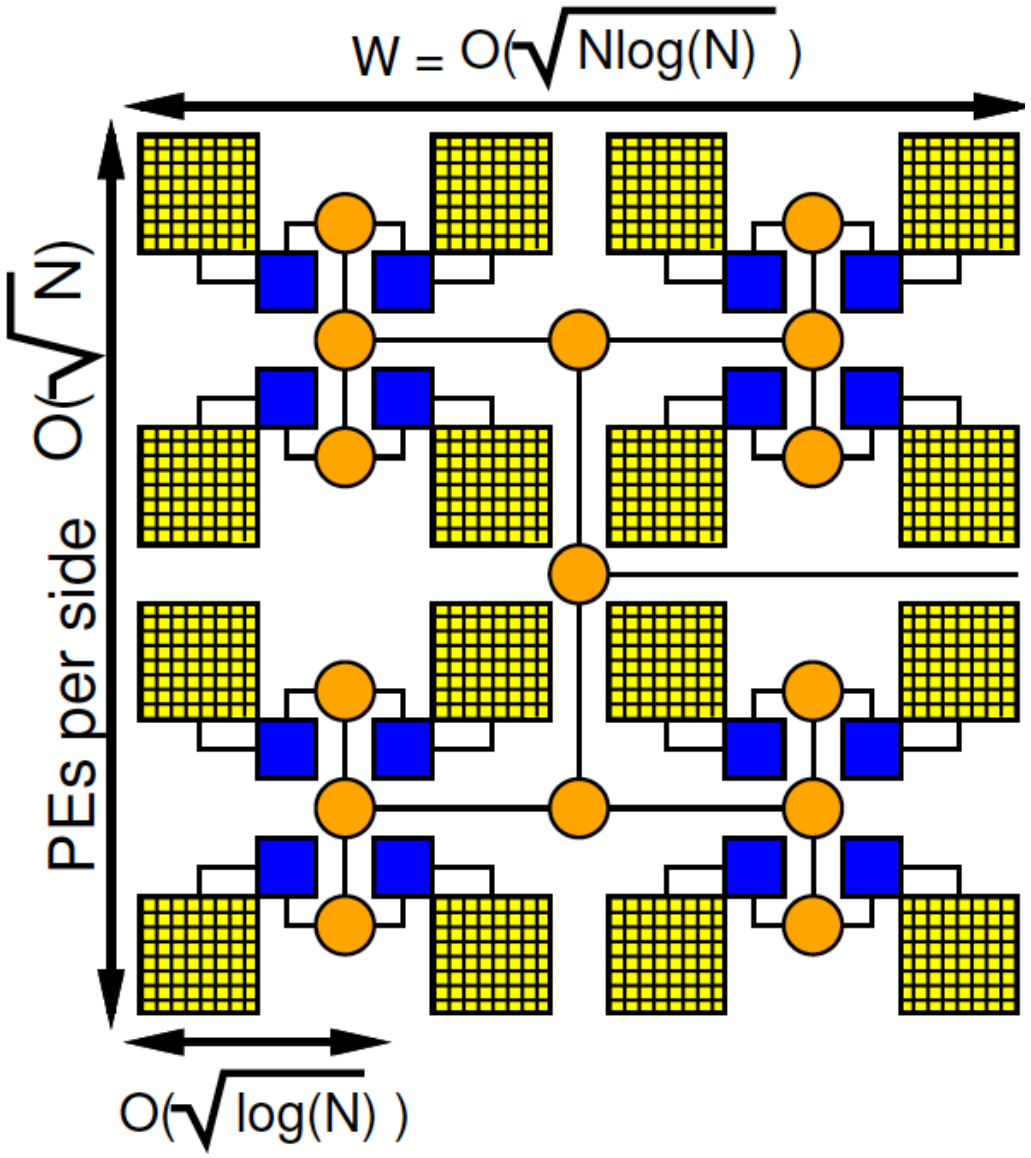
Sequential with Data Locality

- Place for locality -- Rent Partitions
- Store data at endpoints
- Send through network from producer to consumer
- Store location at leaves – O(log(N))
- Build H-Tree to keep area to O(Nlog(N))

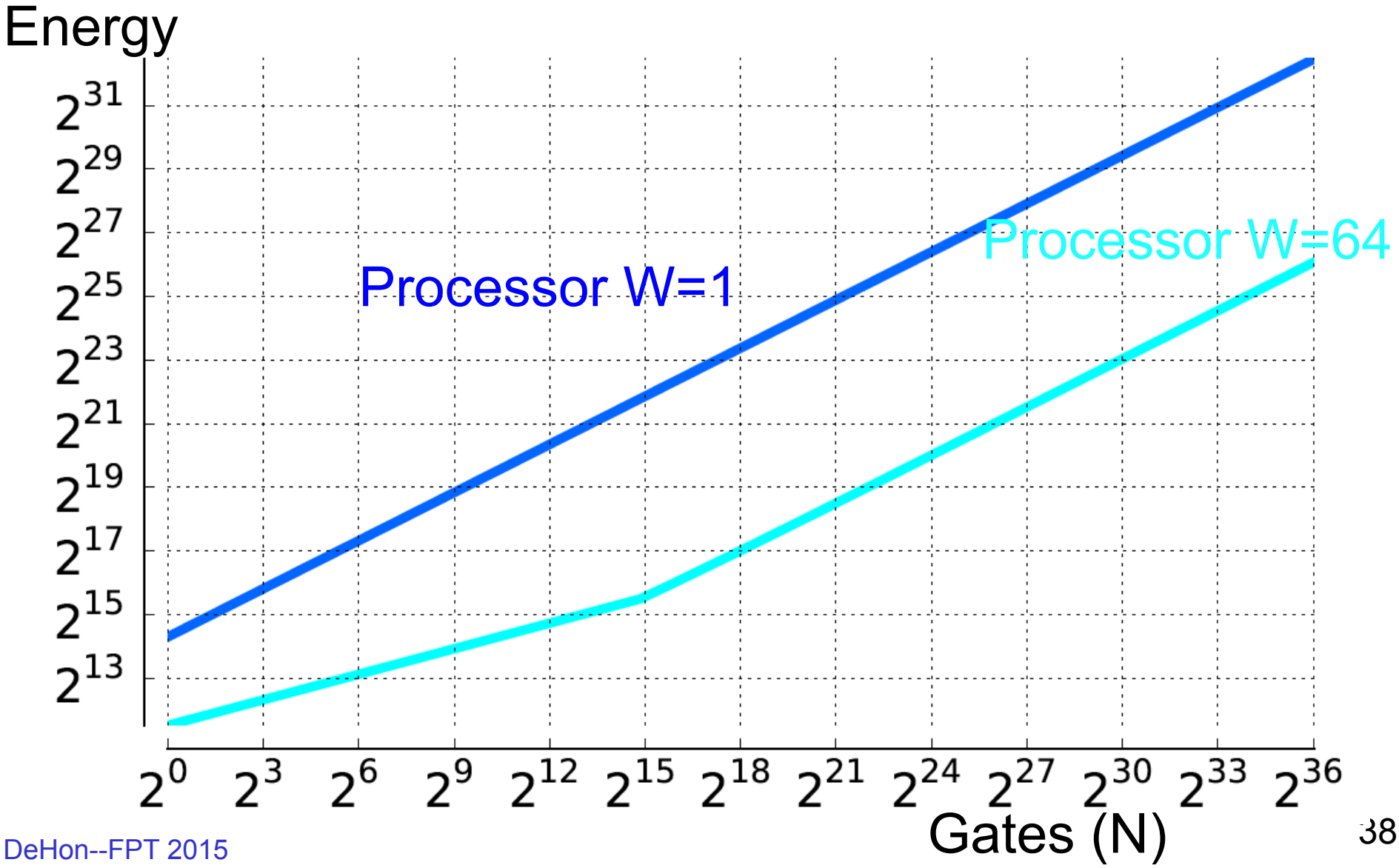


Sequential with Data Locality

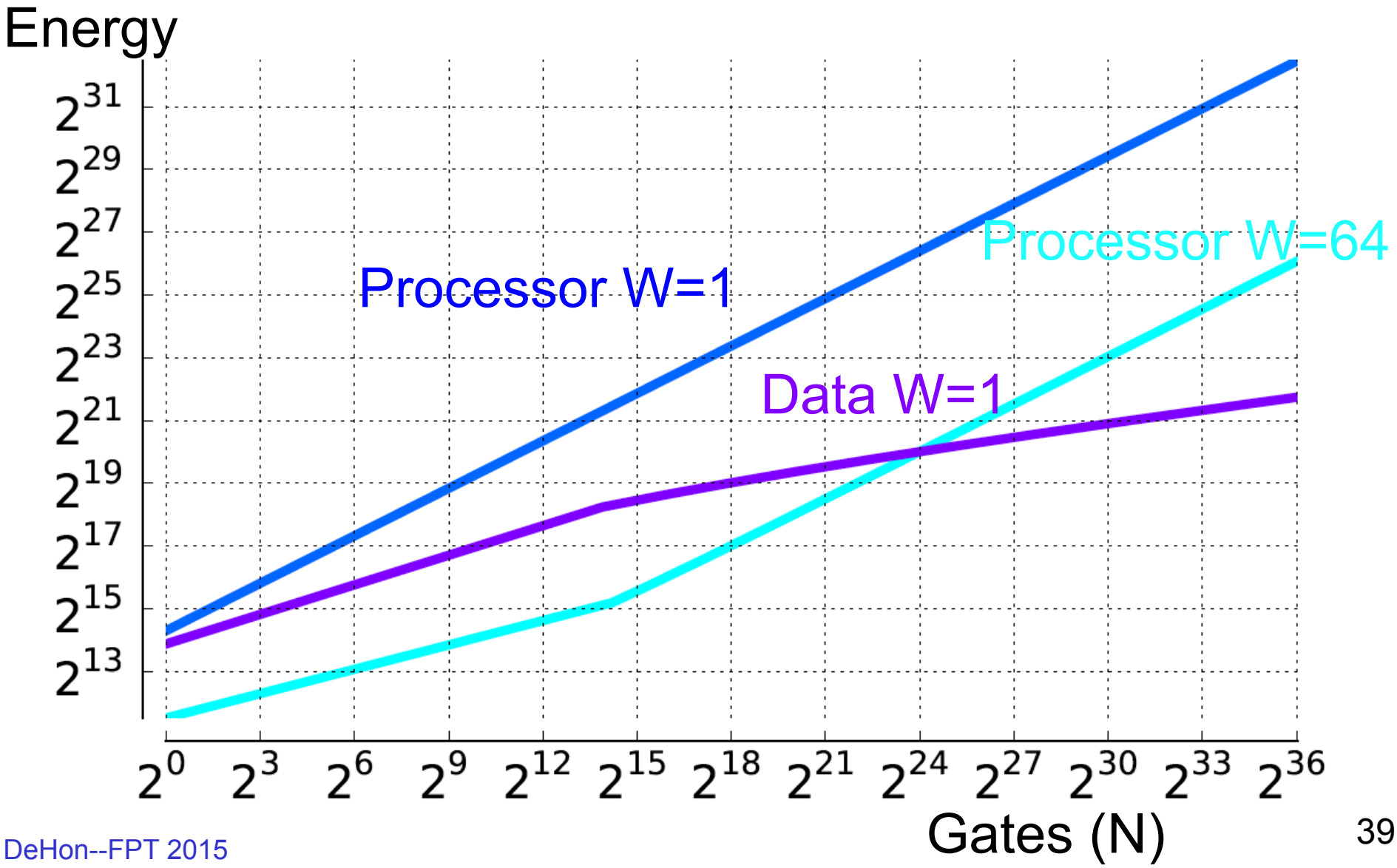
- Area = O(Nlog(N))
- Sending addresses log(N) bits at top
- Signals lower O(1)
- Only send a few over top $O(N^p)$
- $O((\log^{1.5}N)N^{p+0.5})$ for p>0.5
- **Cheaper to send where needed than to central location.**



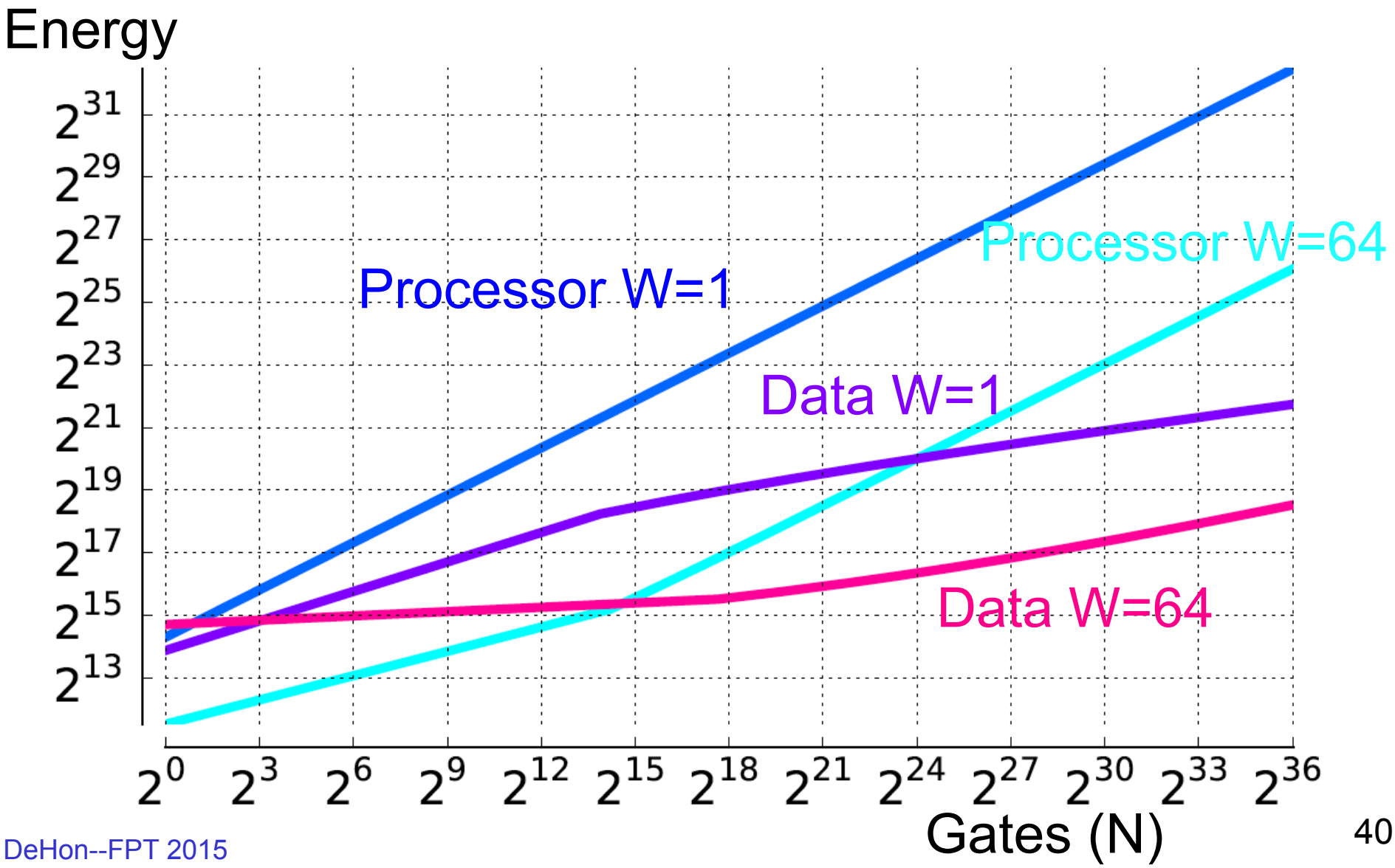
Data Local Compare (p=0.7)



Data Local Compare (p=0.7)

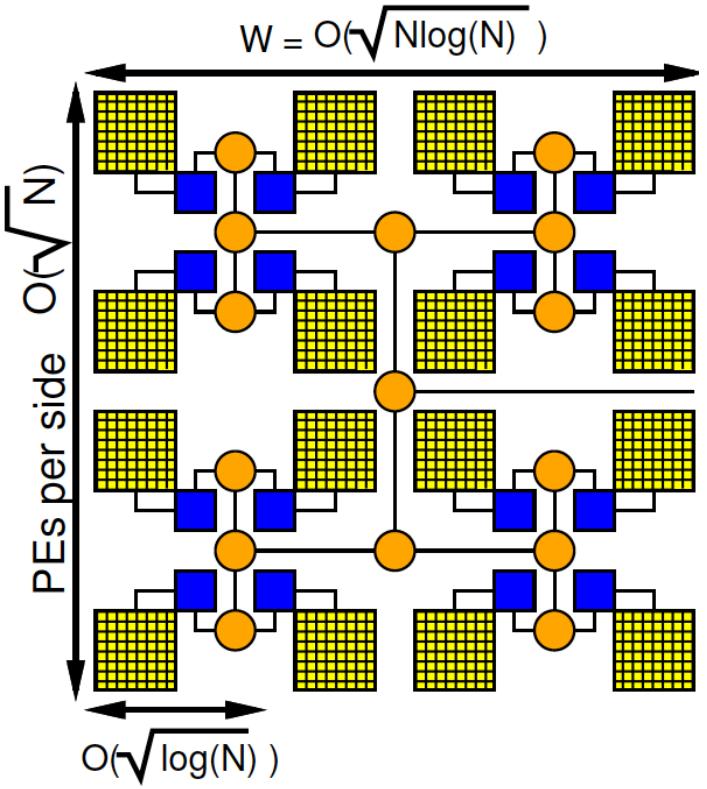


Data Local Compare (p=0.7)



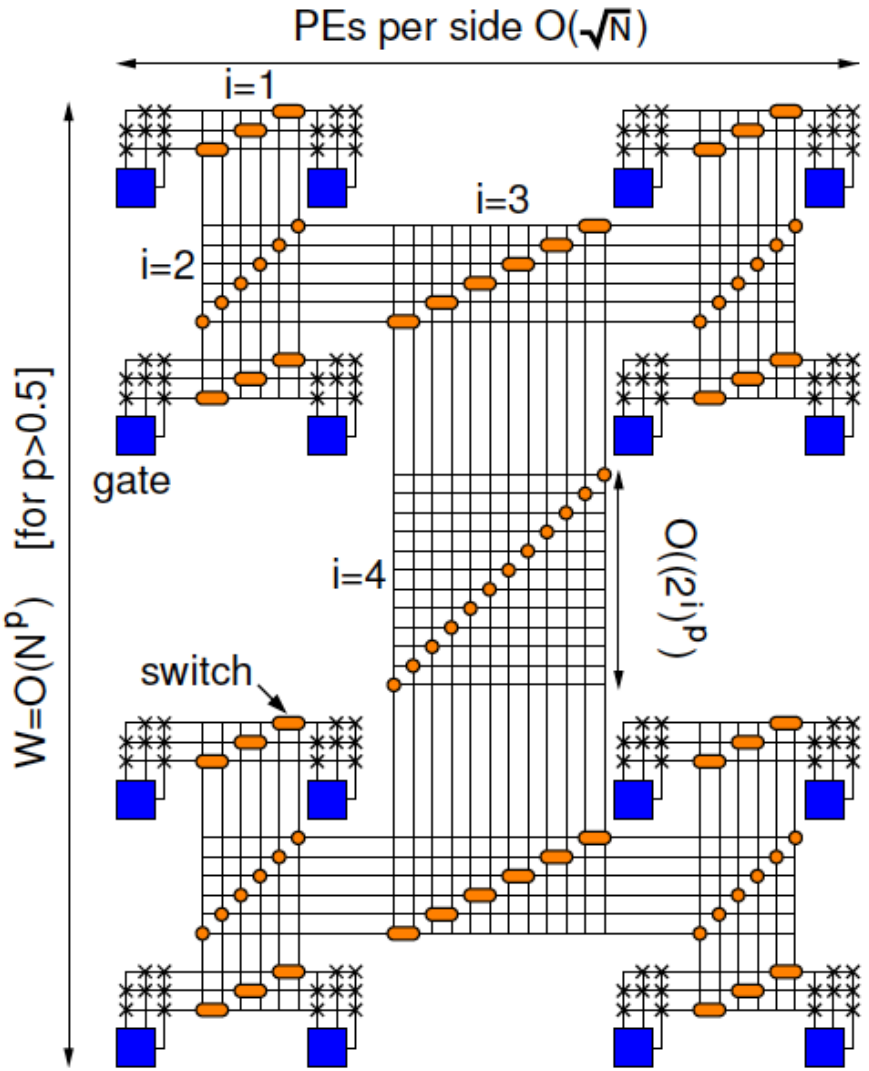
Instruction Locality

- **Problem:** Multiply energy by O(log(N)) to send an address up the tree
- **Opportunity:** store instructions local to switches
  - In tree
- …what an FPGA does!



Fully Spatial (FPGA)

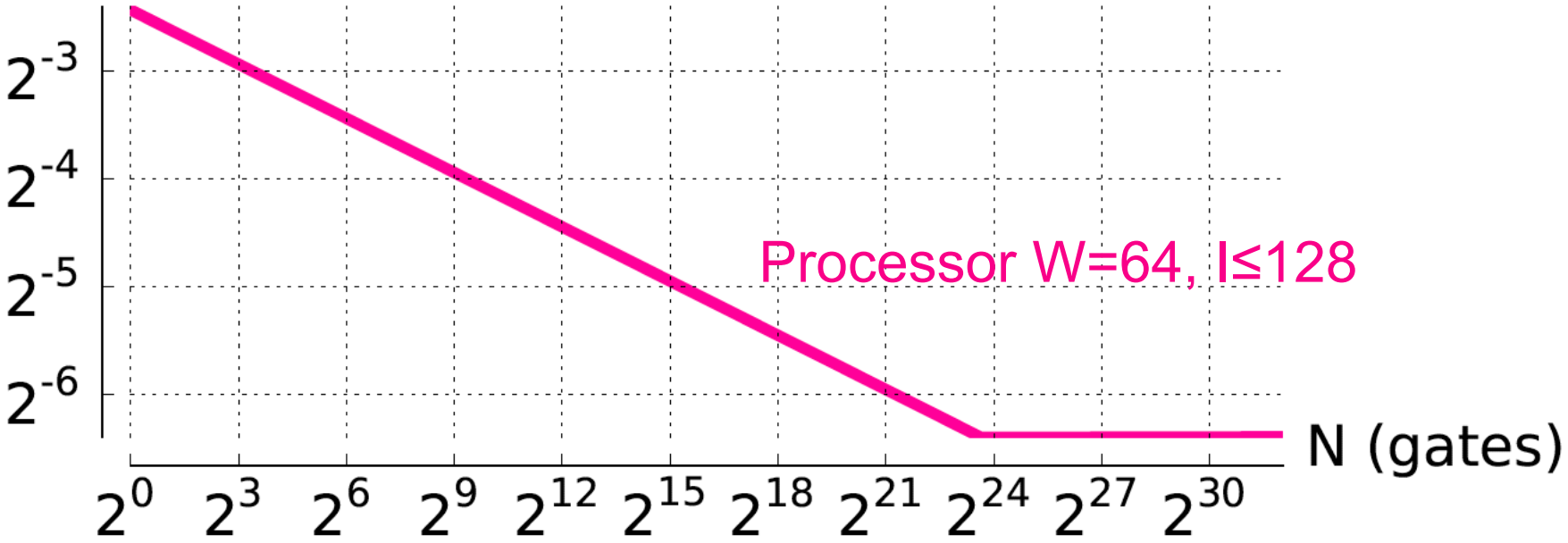
- An FPGA
- Each signal gets own wire
- No addresses
- Configuration local
- Area grows as $O(N^{2p})$ for $p>0.5$
- Energy $O(N^{2p})$ for $p>0.5$
  - $\Theta(N)$ for $p<0.5$



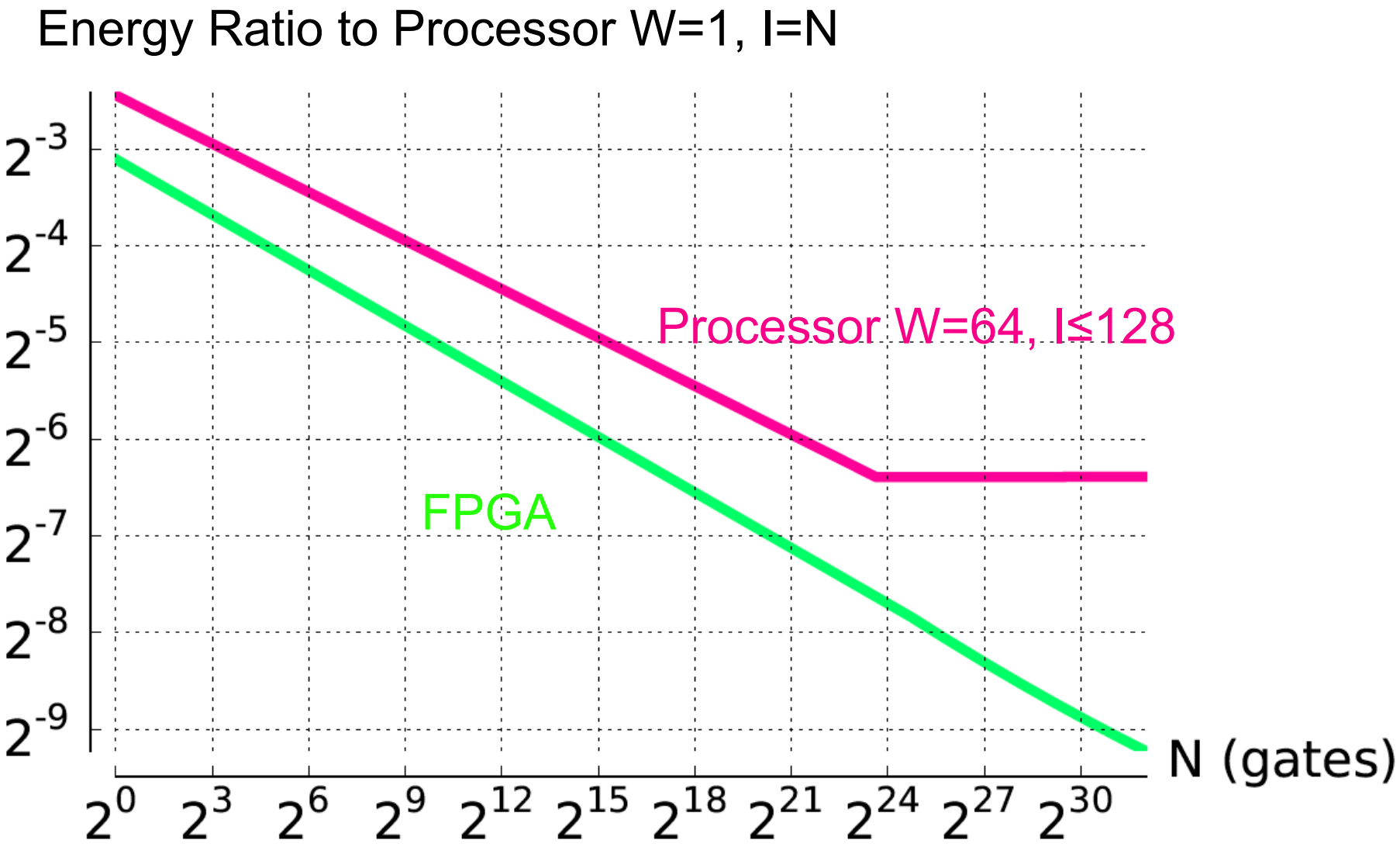
(FPGA-like Architecture)

p=0.7 Compare to Processor

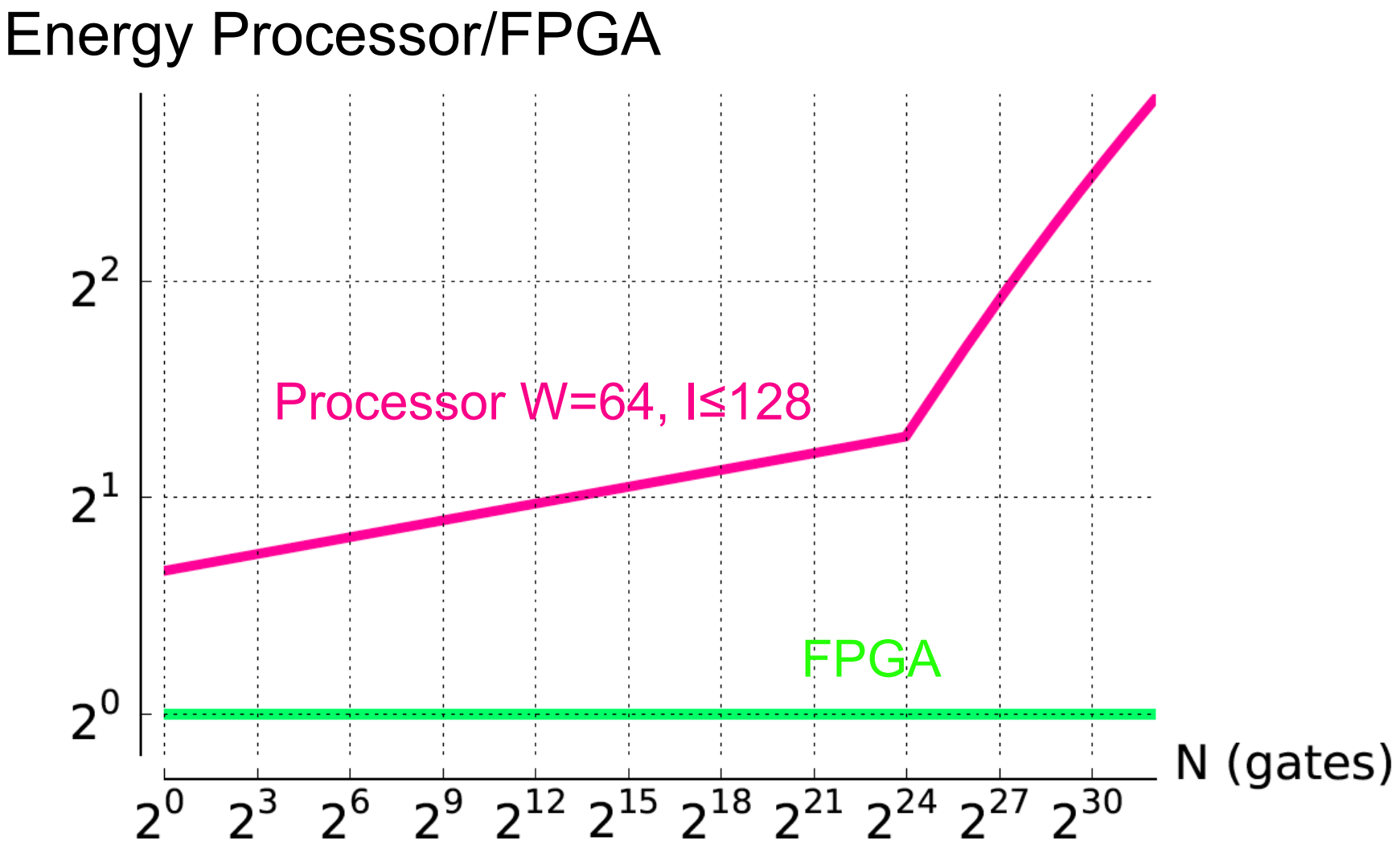
Energy Ratio to Processor W=1, I=N



p=0.7 Compare to Processor



p=0.7 Compare to FPGA



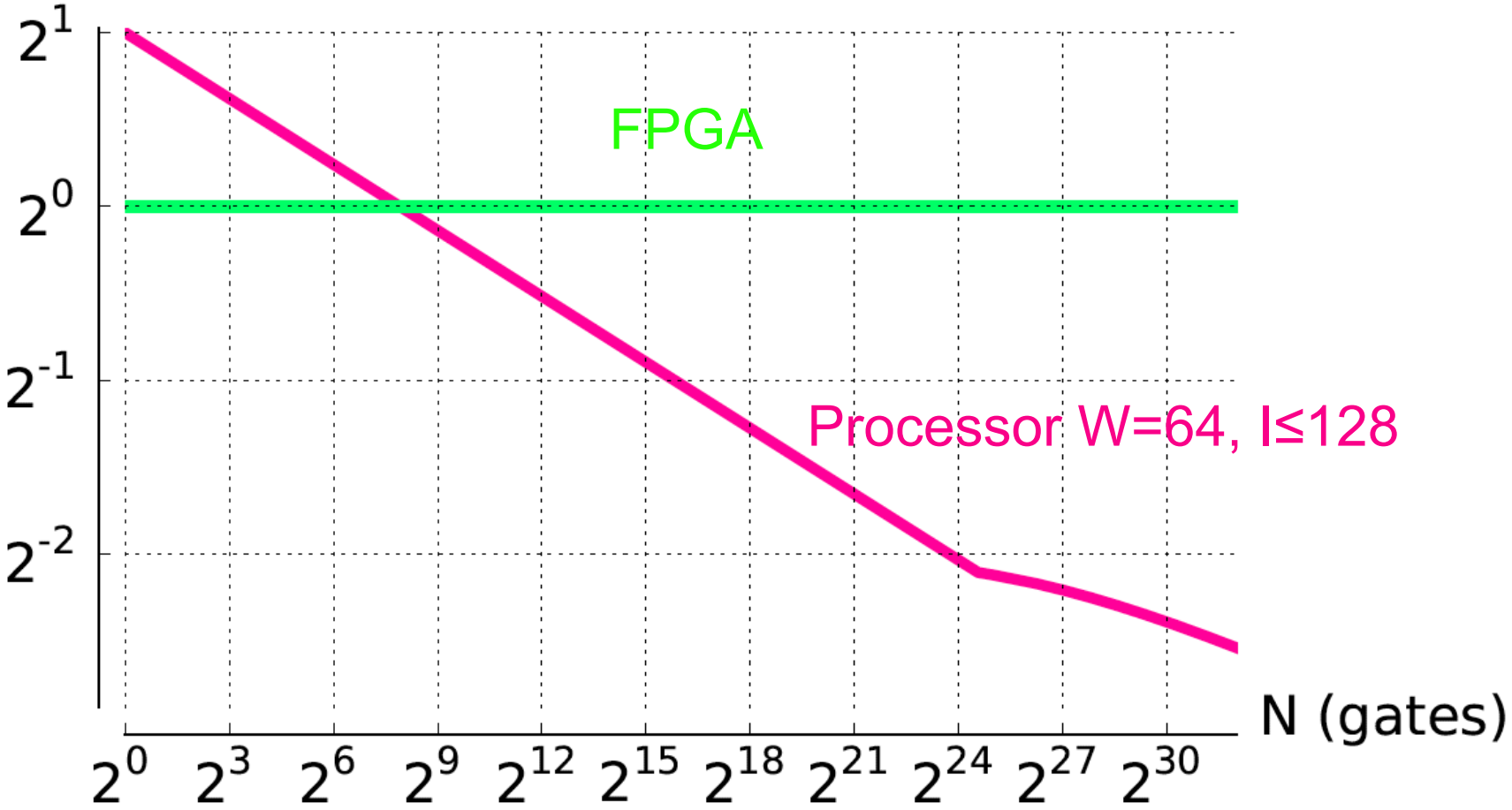
Asymptotic Energy

Org	Any p	p<1.0	1>p>0.5	p=0.5	p<0.5
Processor	$O(N^{1.5}\log^{1.5}N)$	$O(N^{1.5}\log N)$ Description Locality			
FPGA 2-metal			$O(N^{2p})$	$O(N\log^2 N)$	$\Theta(N)$

Note break at p=0.75

p=0.8 Compare to FPGA

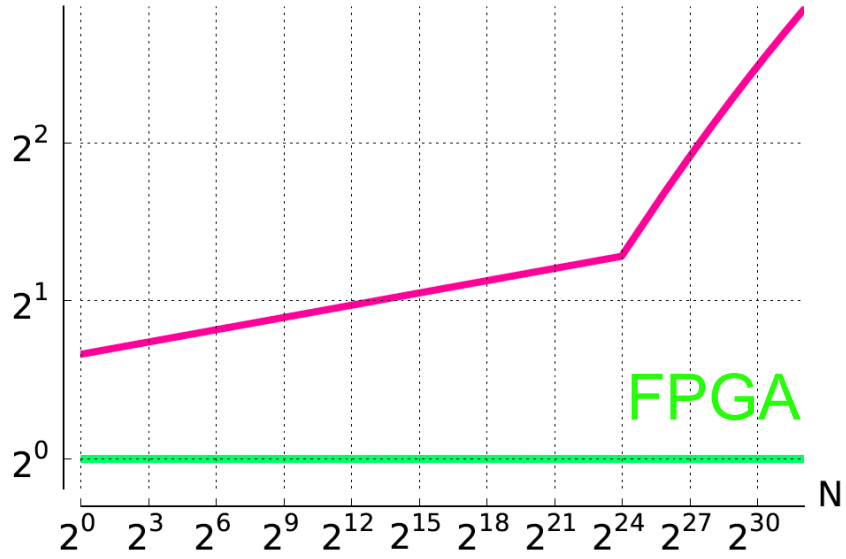
Energy Processor/FPGA



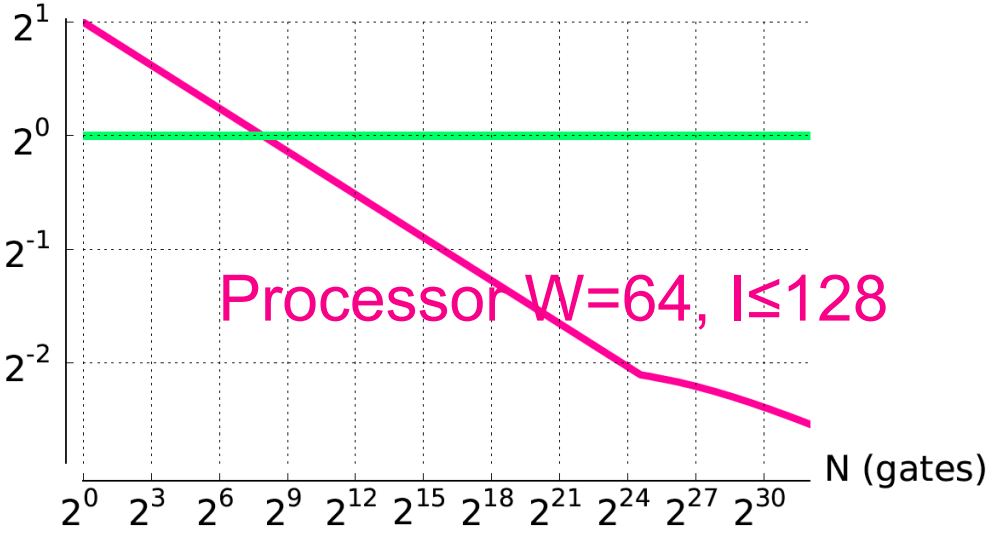
Compare around p=0.75

- Breakpoint at p=0.75

Energy Processor/FPGA



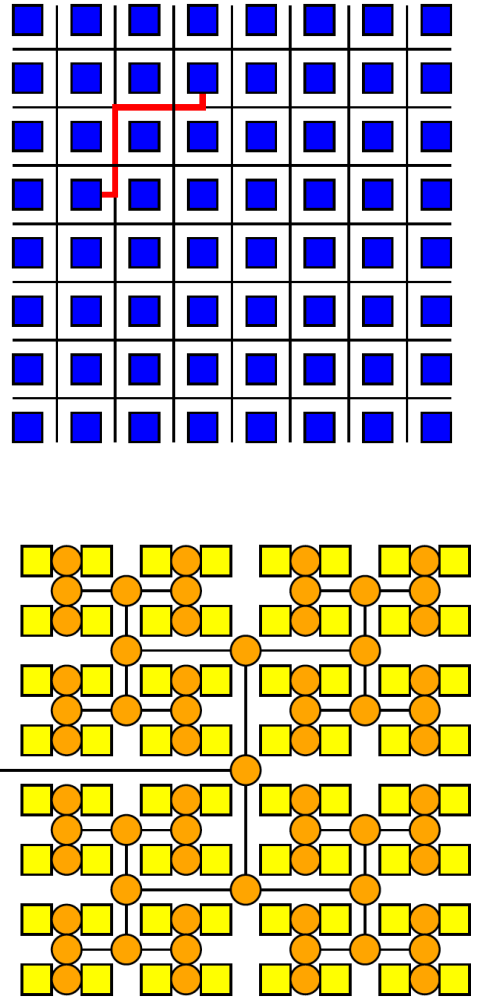
p=0.7



p=0.8

Intuition

- Given a good spatial layout
  - It is cheaper to transmit the result of a gate to its well-placed consumers
    - Fixed metal layers: average wire length
      - p<0.5: O(1)
      - p<0.75: $O(N^{2p-1}) < O(N^{0.5})$
      - p>0.75: $O(N^{2p-1}) > O(N^{0.5})$
  - Than to
    - Fetch inputs from a large central memory
      - $O(N^{0.5})$



Architecture Outline

- Rent's Rule and VLSI Complexity
- Memories
- Central Processor
- Spatial Locality
  - Data
  - Instruction
- Wire Sharing

Wiring Dominates

- **Problem:** When p>0.5 wiring dominates area

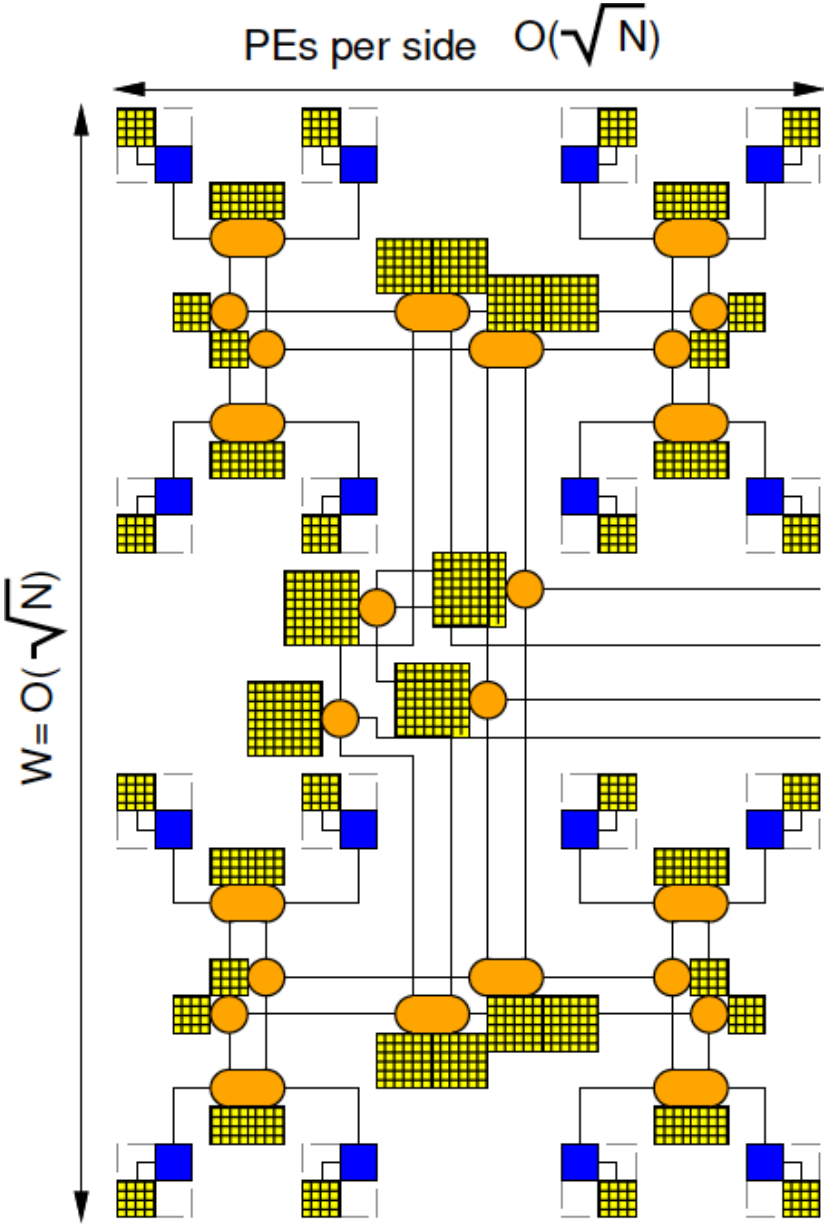
  → force longer wires

- **Opportunity:** Share wires

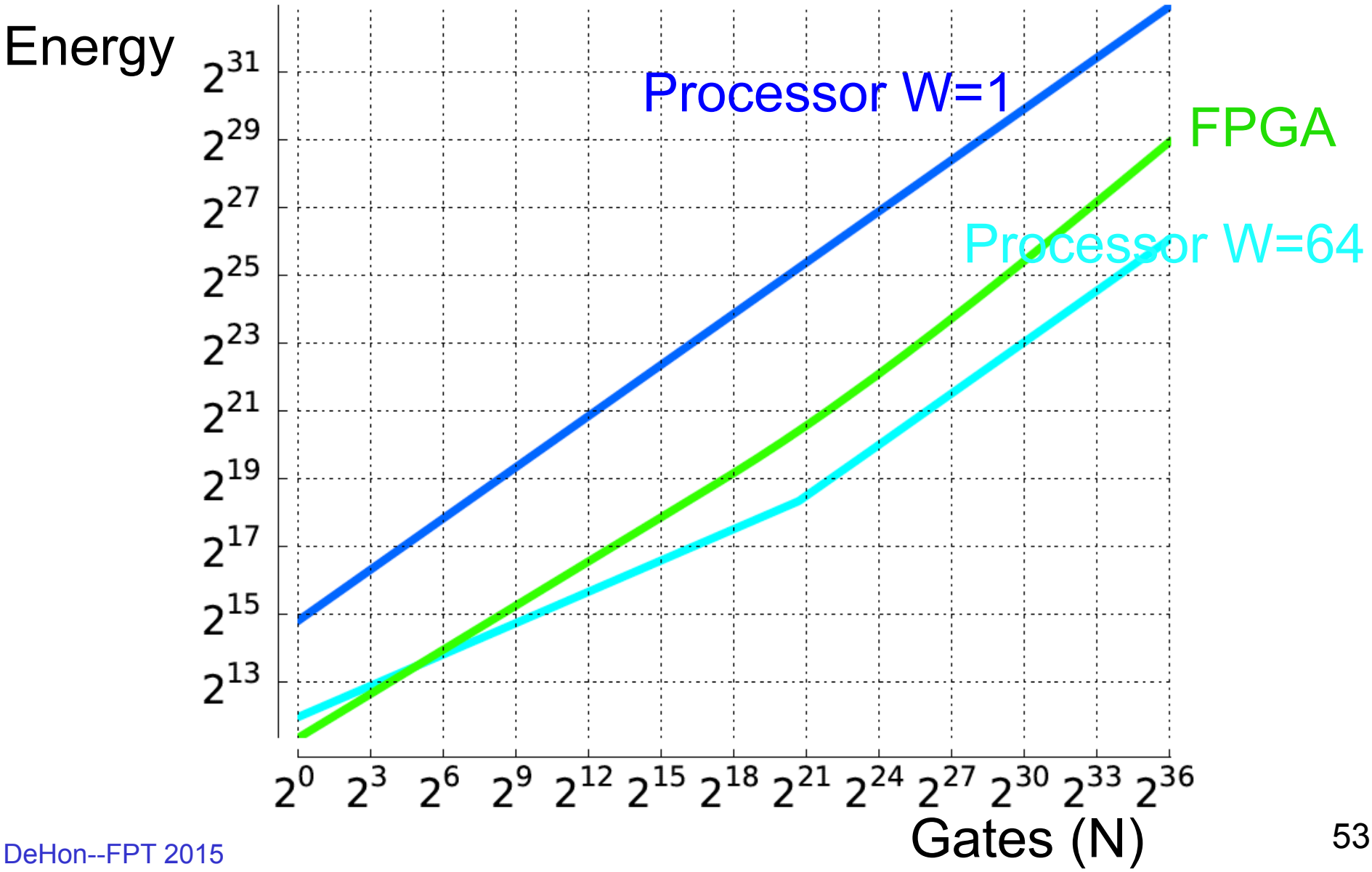
  But also, keep instructions local to switches

Instructions Local to Switches

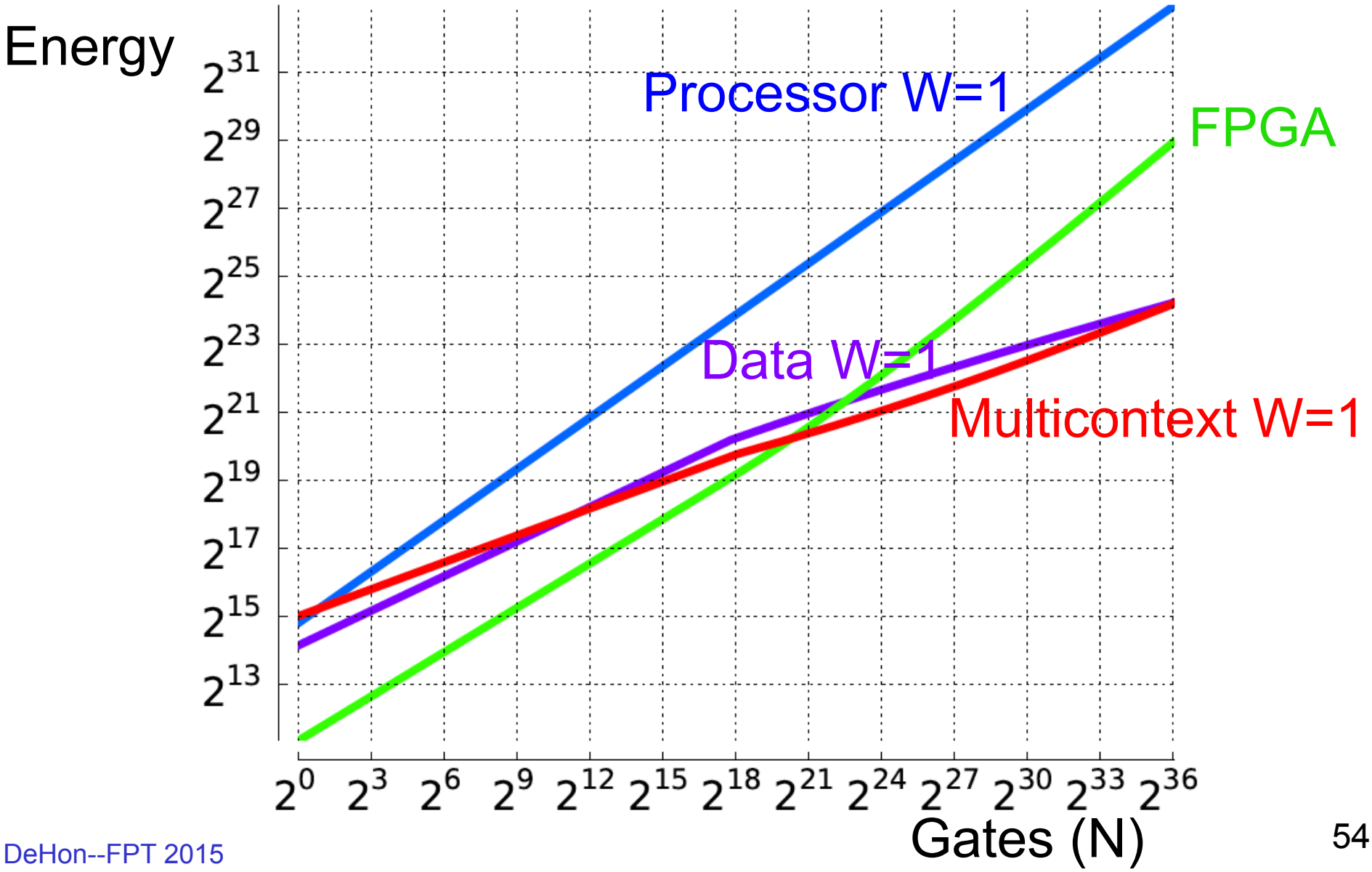
- Constant metal
- Build p<0.5 tree
- Store bits local to each tree level
- Read out of memory there
- Bits/switch differs with tree level
- Signal on wire dominates reads
- $O(N^{p+0.5})$ for p>0.5



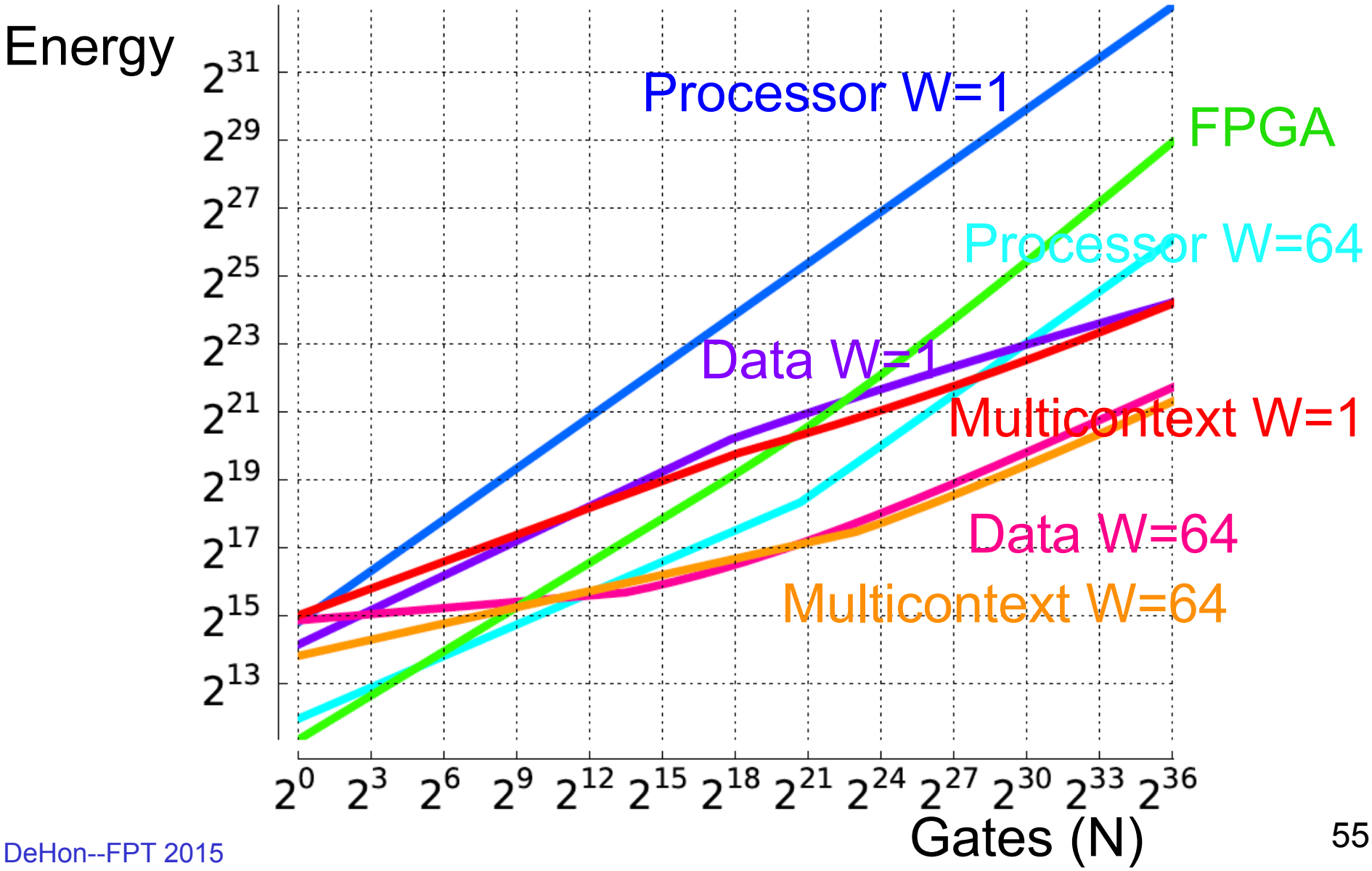
Instructions Local (p=0.8)



Instructions Local (p=0.8)



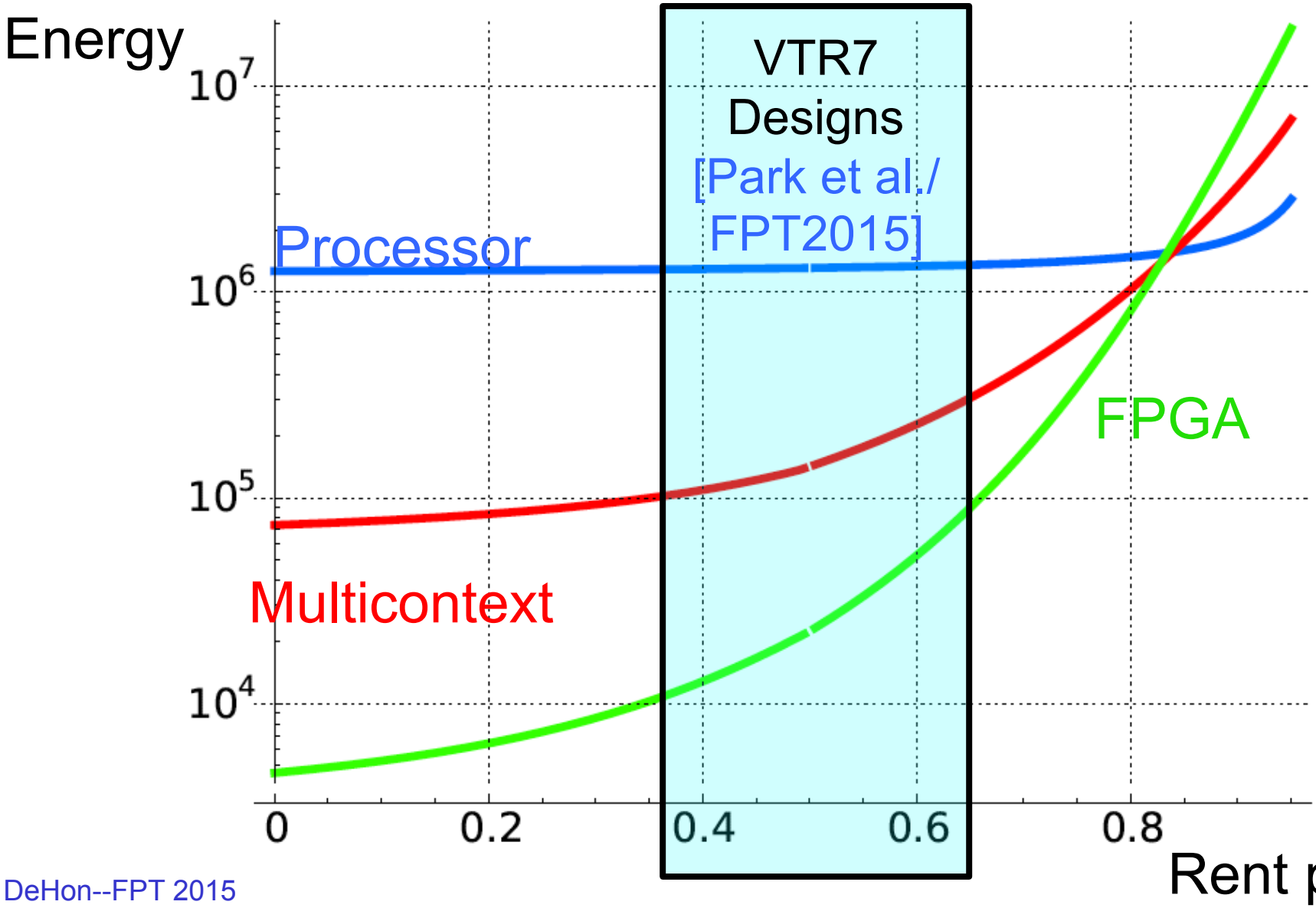
Instructions Local (p=0.8)



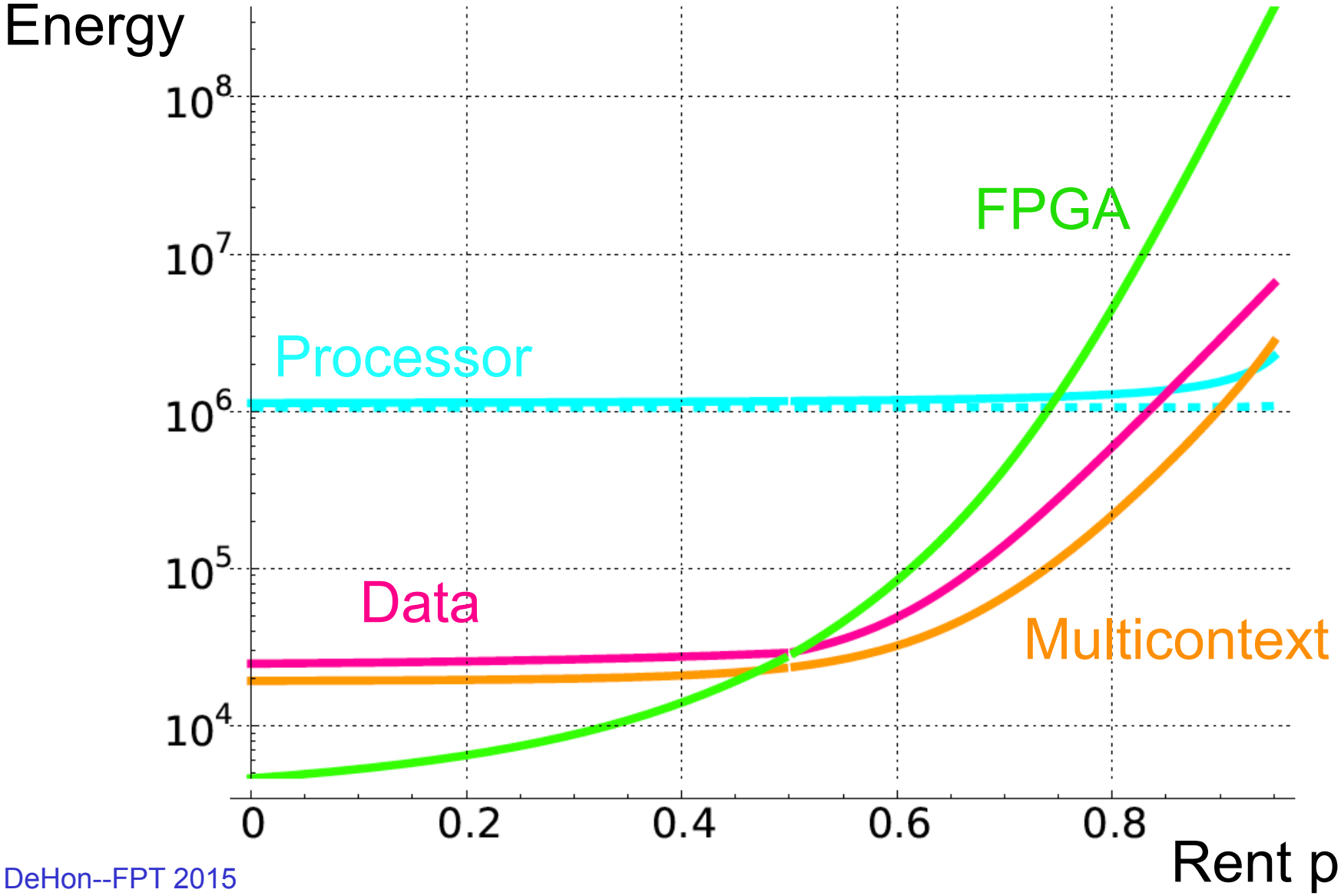
Results: Energy

Org	Any p	p<1.0	1>p>0.5	p=0.5	p<0.5
Processor	$O(N^{1.5}\log^{1.5}N)$	$O(N^{1.5}\log N)$ Description Locality			
Data Locality (Packet Switch)			$O(N^{p+0.5}\log^{1.5}N)$	$O(N\log^{2.5}N)$	$O(N\log^{1.5}N)$
FPGA 2-metal			$O(N^{2p})$	$O(N\log^{2}N)$	$\Theta(N)$
Multicontext			$O(N^{p+0.5})$	$O(N\log N)$	$\Theta(N)$

Compare vs. p, W=1, N=2^{19}

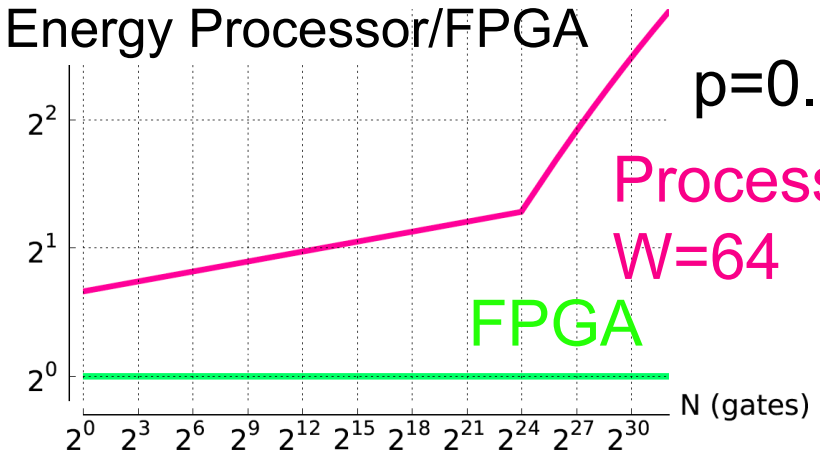


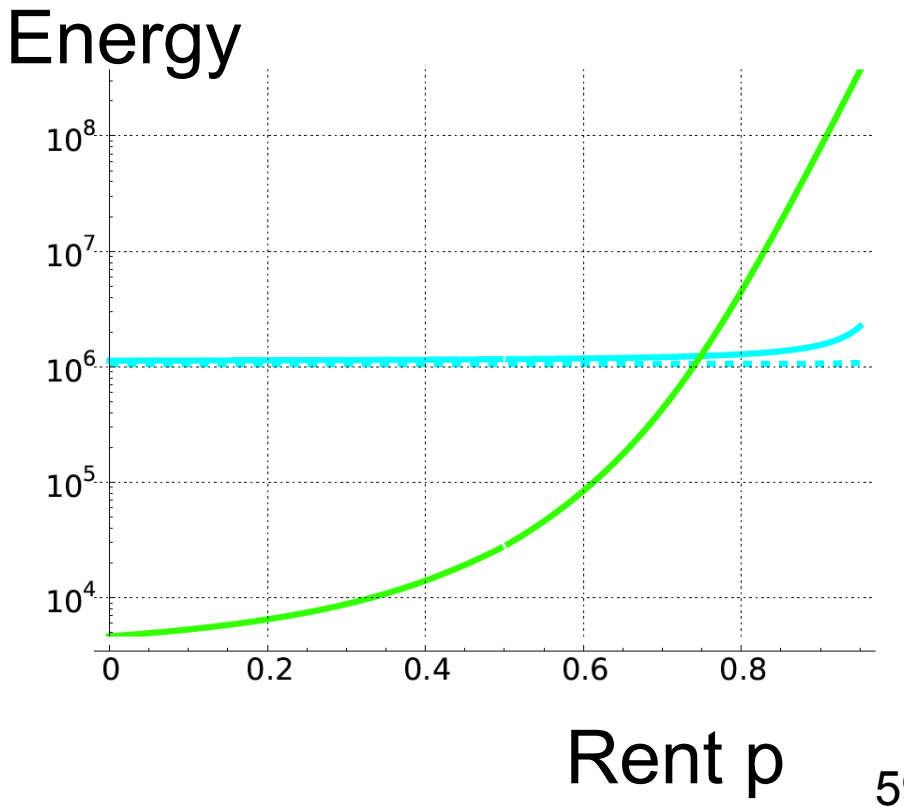
Compare vs. p, W=64, N=2^{24}



Architecture

- There are architectural energy advantages to FPGAs over processors





$$E \propto \alpha C V^2$$

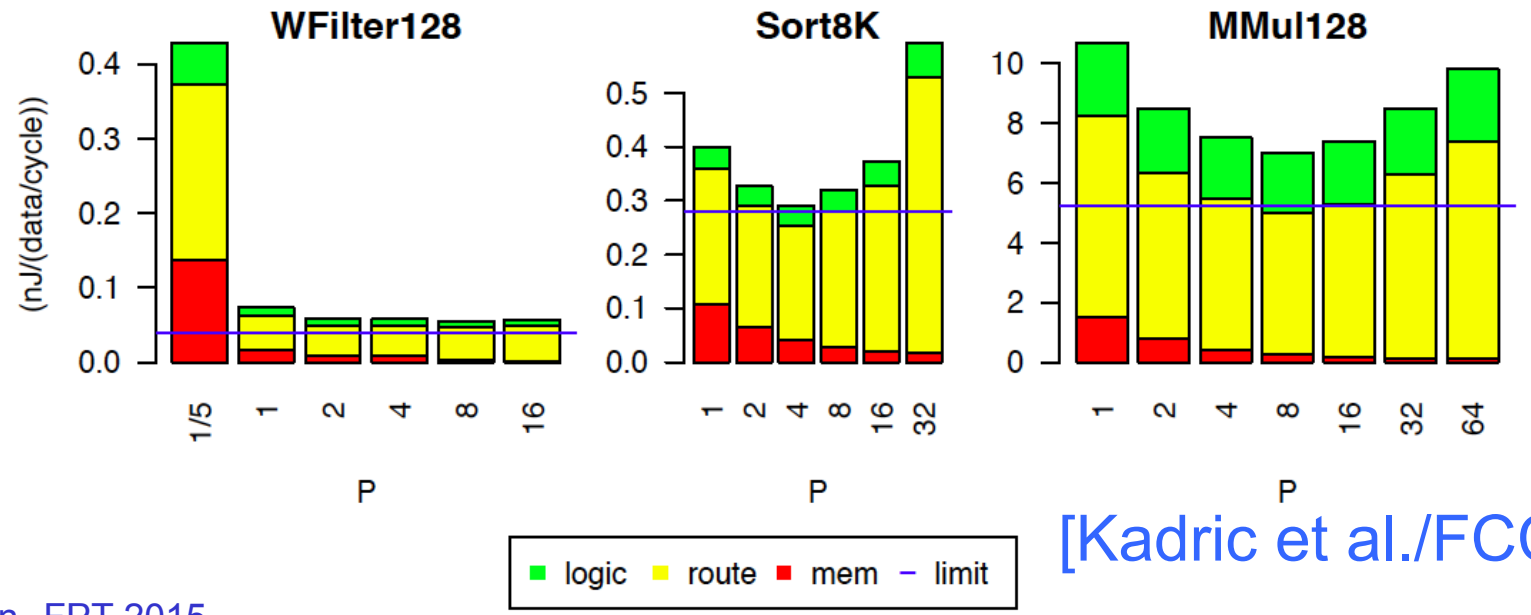
Optimizing FPGAs and Applications

Tune Parallelism

- For many, regular designs we can tune the parallelism in the **implementation** on top of the FPGA
- Parameterize number of PEs
- Store data in Embedded Memories
- Tune to balance Memory vs. Interconnect costs

Optimal Level of Parallelism

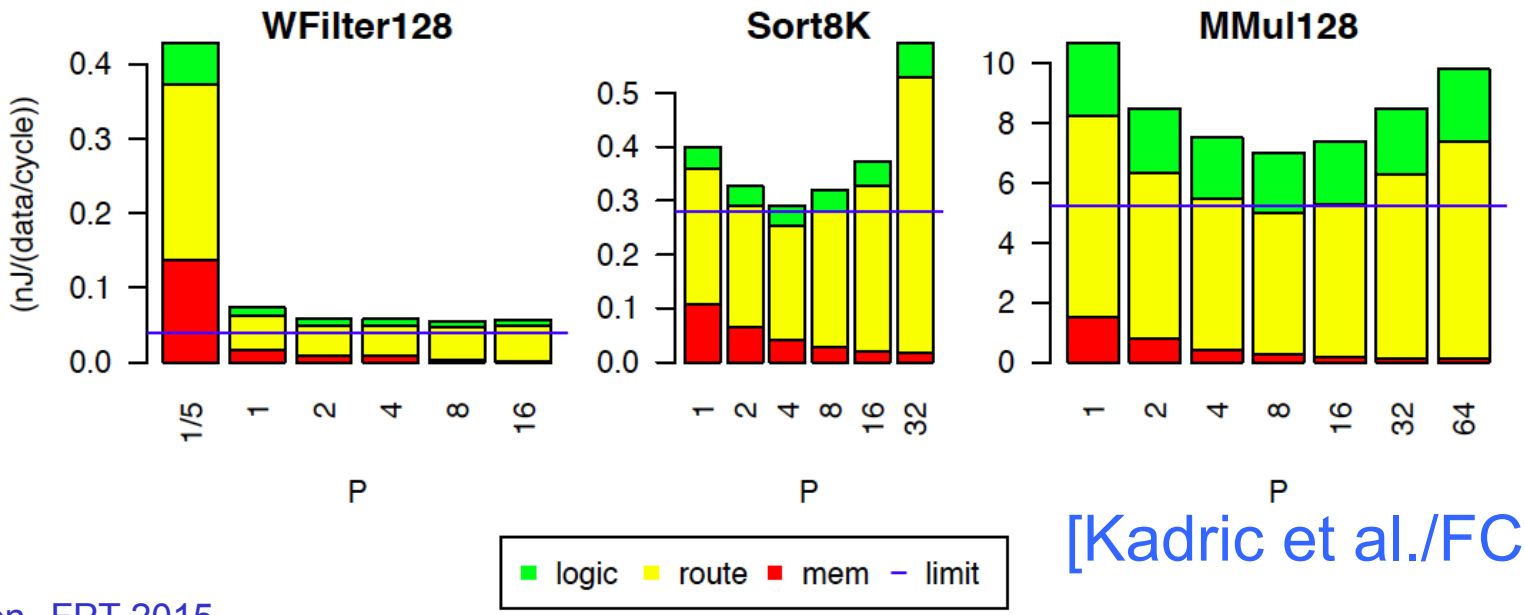
- For regular designs
  - Build common Operator
  - Share amongst logical operations
  - Use embedded RAMs to hold state of multiple operators



[Kadric et al./FCCM2014]

FPGA Energy Efficiency

- When data memory dominates
- Question is not:
  - How efficient is FPGA vs. ASIC? … but
  - How efficient is FPGA memory vs. ASIC memory?



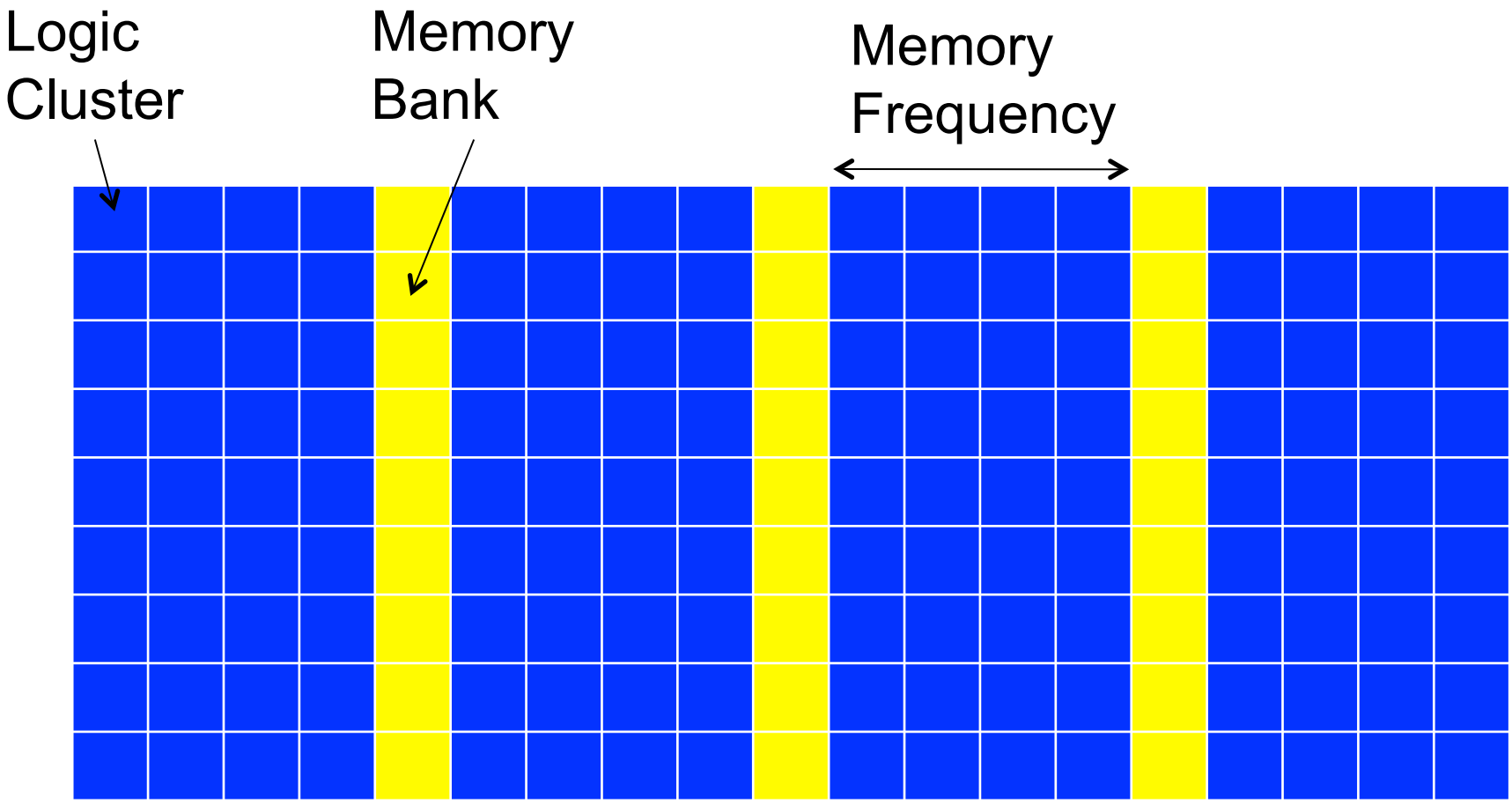
[Kadric et al./FCCM2014]

FPGA Memory Energy Optimization

- Energy inefficiency from
  - Memory blocks too large
  - Memory blocks too frequent, infrequent
  - Activating memory unnecessarily
- Can get within factor of 2
  - Appropriate distribution of blocks
  - Efficient memory banking

[Kadric et al./FPGA2015]

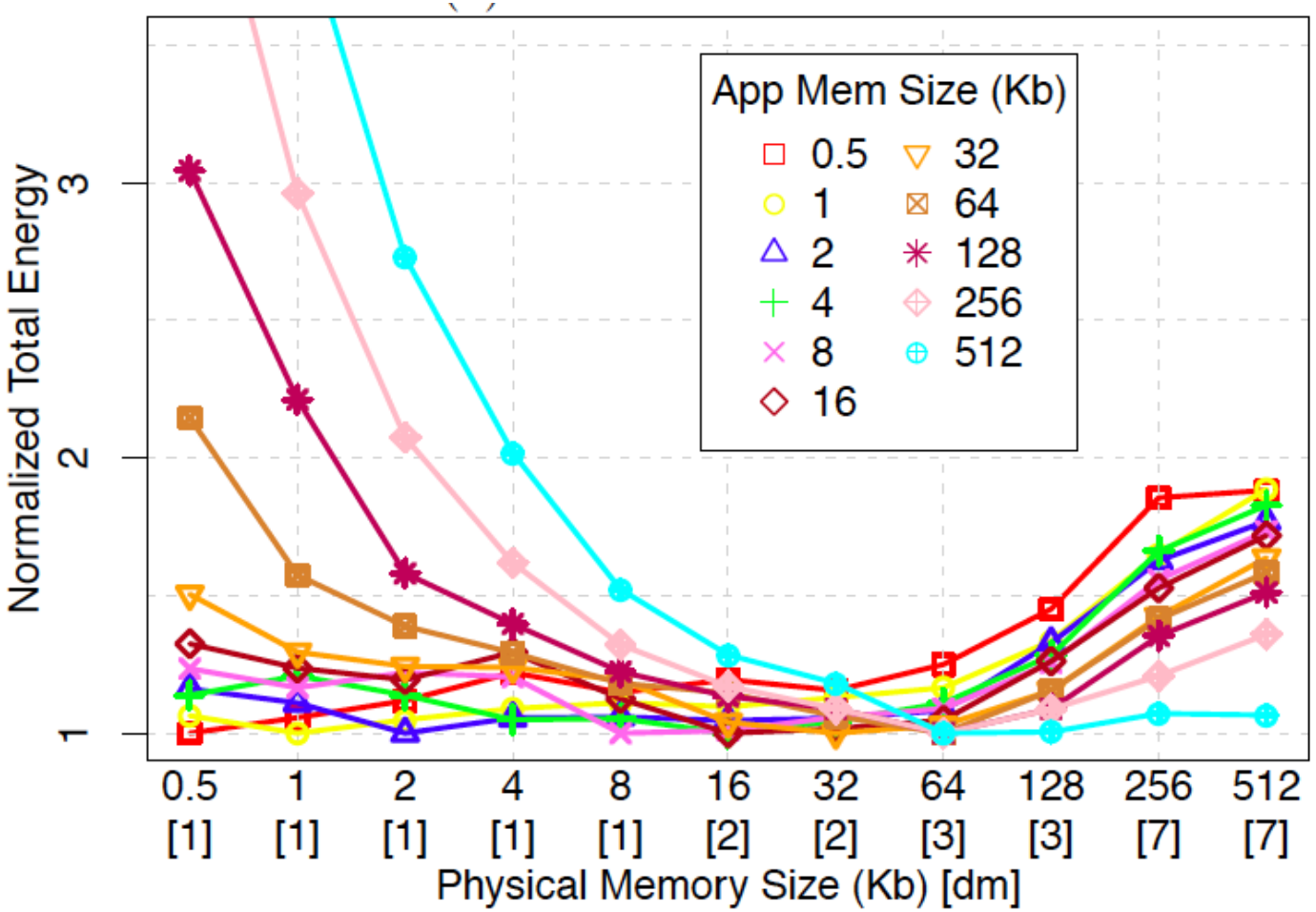
Modern FPGA



- Memory Bank size? (Distribution of sizes?)
- Memory Bank frequency?

Matrix Multiply, Single Memory

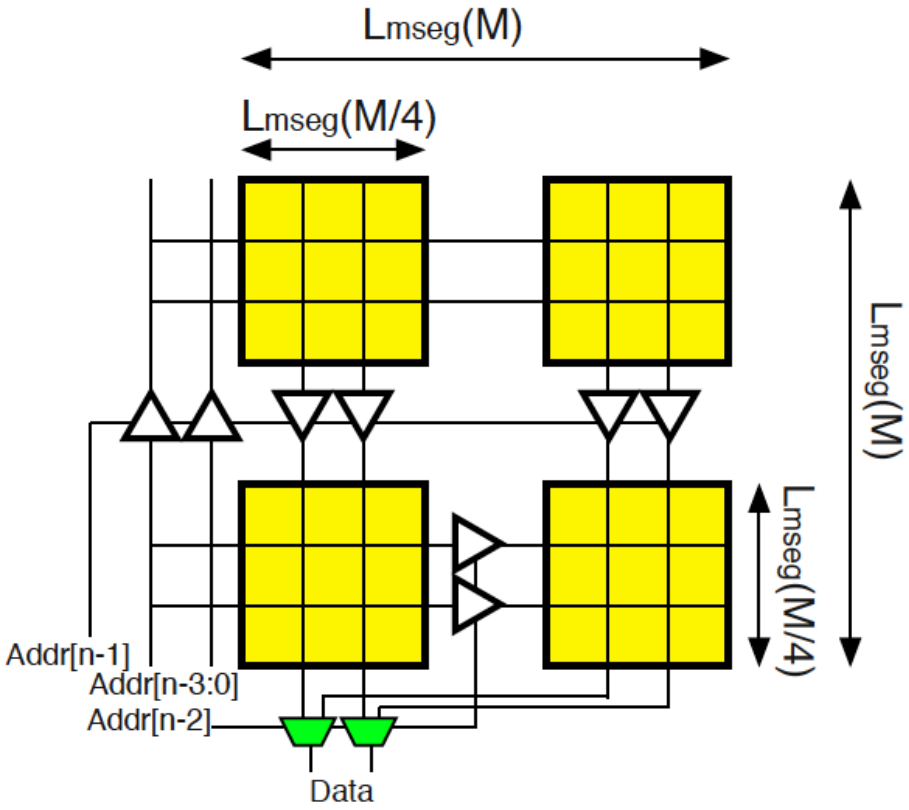
Energy normalized to best match



[Kadric et al./FPGA2015]

Internal Banking

- Recall memory energy scale as $O(M^{0.5})$
- Continuous Hierarchy Memory
  - Internal Banking
- Only pay as much energy as need

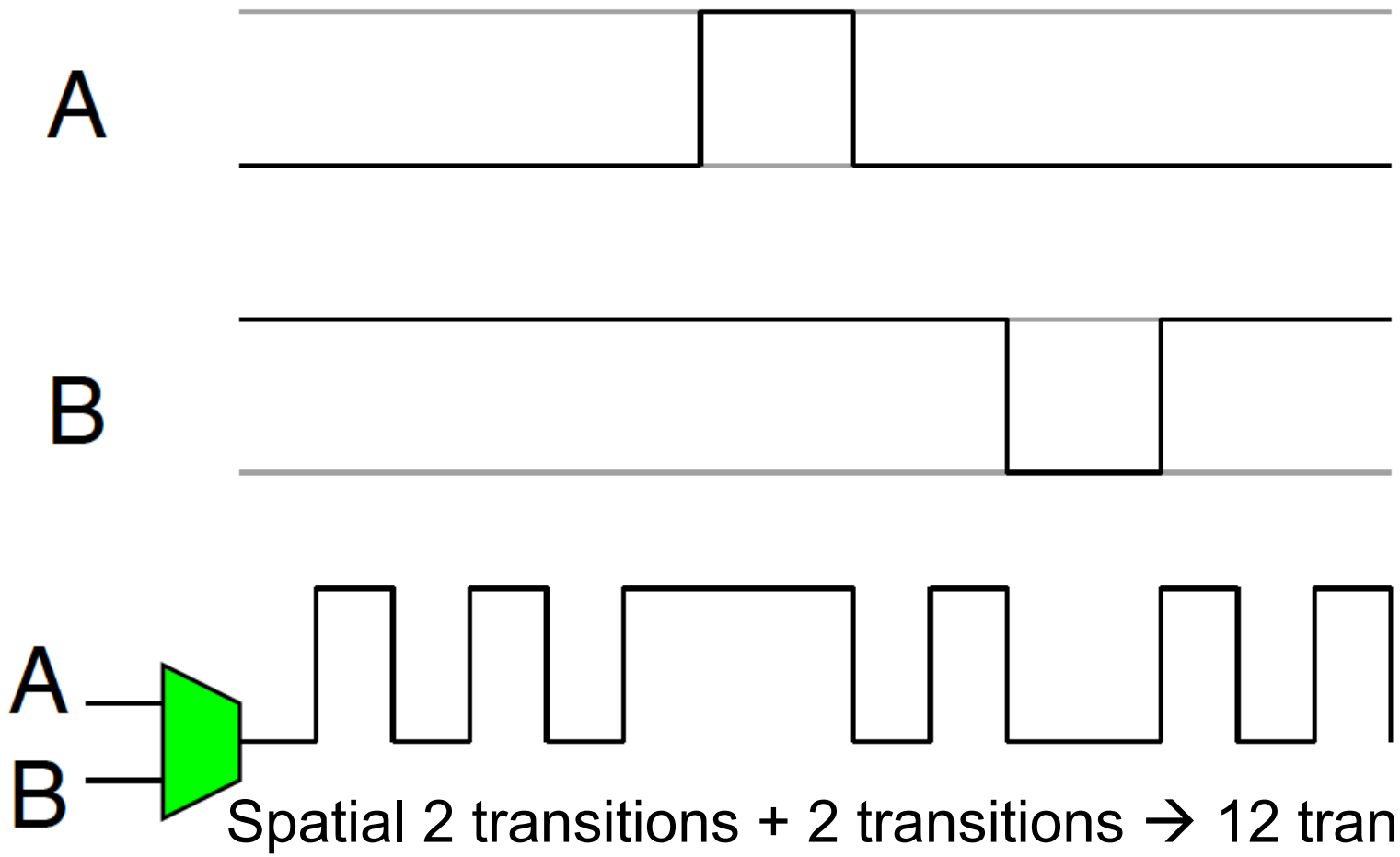


[Kadric et al./FCCM2014+FPGA2015]

$$E \propto \alpha C V^2$$

Information (Switching Activity)

Time Multiplexing Decorrelates Signals

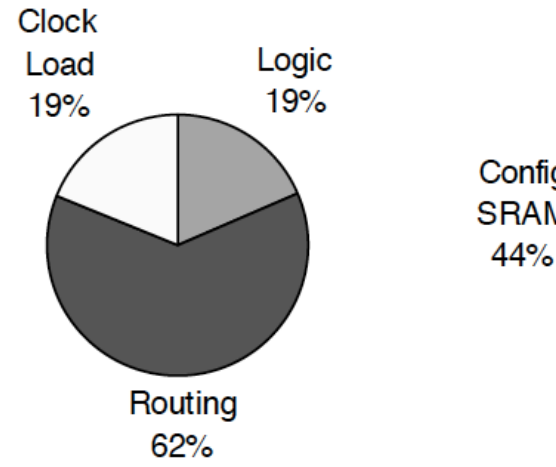


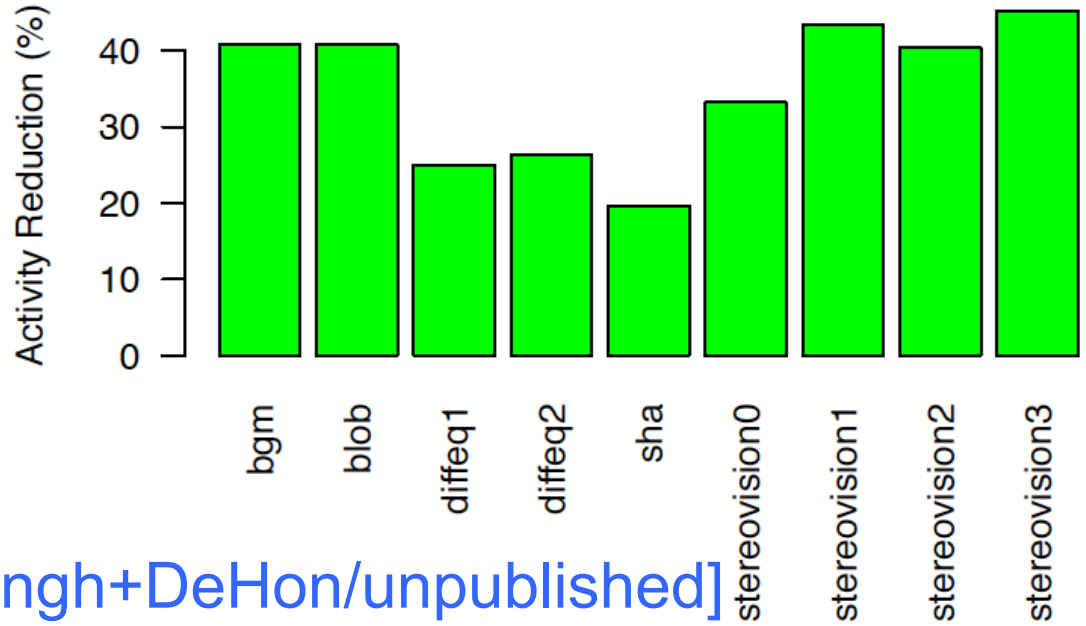
Spatial 2 transitions + 2 transitions → 12 transitions

[Park et al./FPT2015]

Encode to Reduce α

- Dominant FPGA energy
  - Is energy in wires
  - especially in p>0.5 cases
- Encode to reduce switching





[Tuan et al./ FPGA 2006]

[Singh+DeHon/unpublished]

Outline

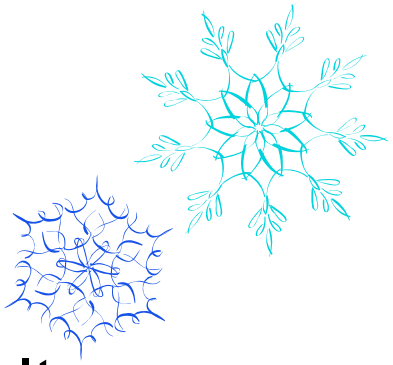
- Energy: why care?
- Teasers
- Architecture
- Implication for
  - Applications
  - FPGA Architecture
- Information (Coding)
- Variation and Aging

$$E \propto \alpha CV^2$$

Variation and Aging

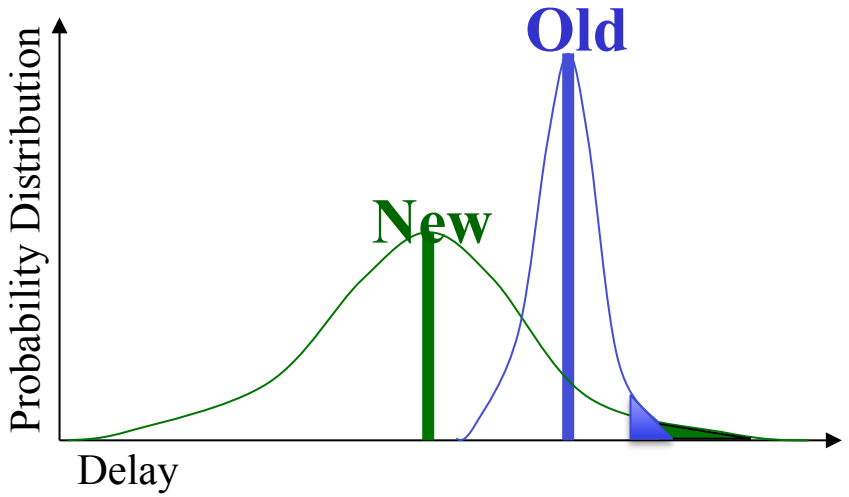
Voltage Challenge

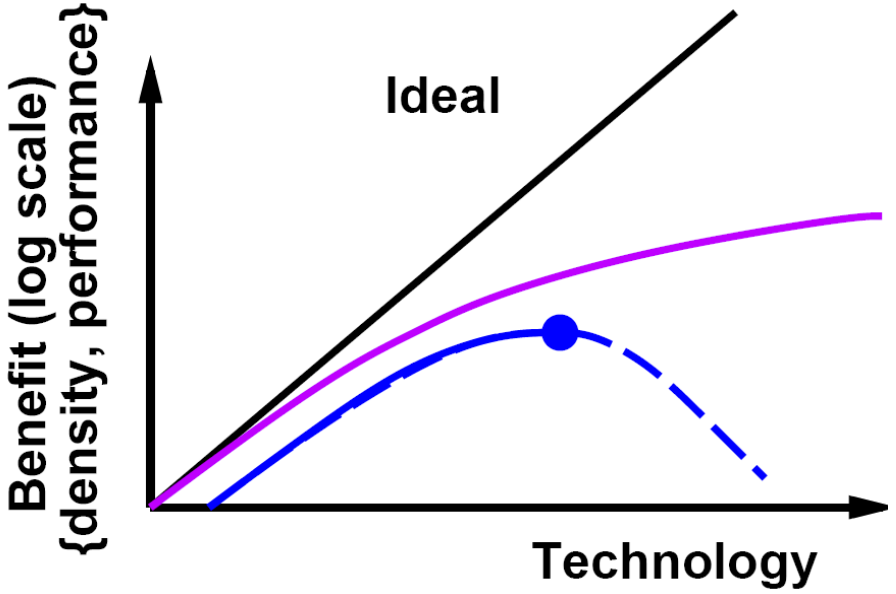
- End of Dennard Scaling
  - Subthreshold slope prevents linear reduction of Voltage with feature size
- Variation
  - Devices no longer “identical”
  - Typically accommodate with higher voltage
- Aging
  - Devices change
  - Typically accommodate with higher voltage



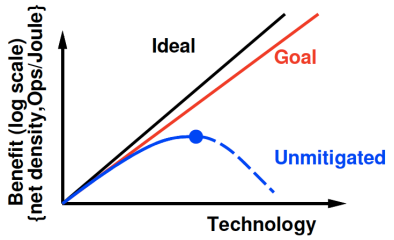
Inflection Point Collision Defeat Scaling

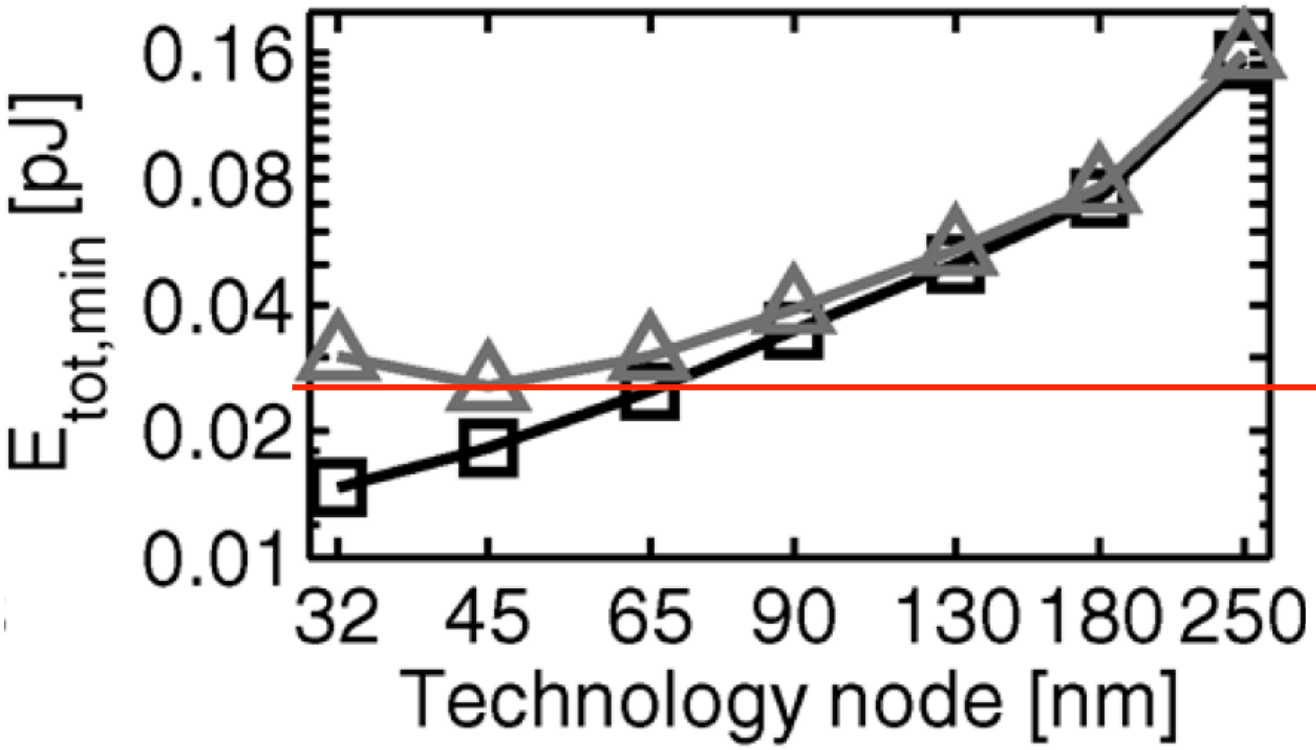
- Spend energy for reliability
  - Margins
  - Replication
- Reduce or eliminate net benefits of feature size reduction





Variation threatens E/Op reduction





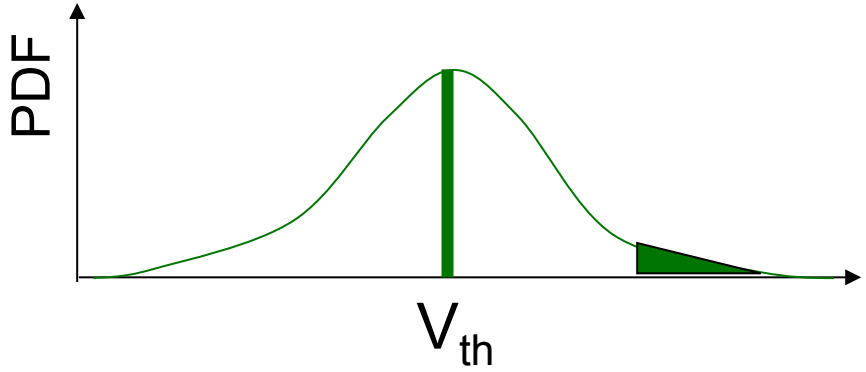
Black nominal

Grey with variation

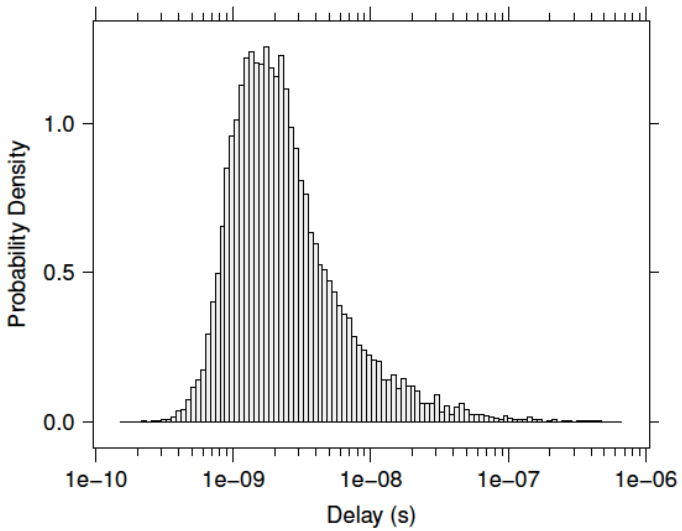
Min-Energy for multiplication (typically subthreshold)

[Bol *et al*., IEEE TR VLSI Sys 17(10):1508—1519]

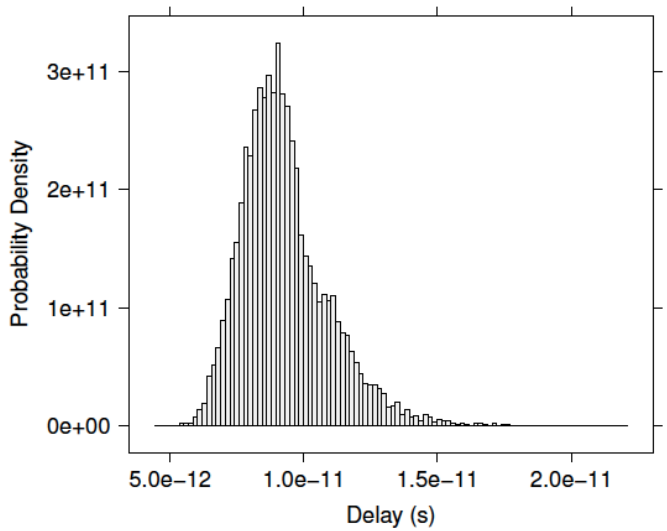
Driven by Tails



- High margins driven by uncommon tails
  - Most devices much better
- Large device count → sample further into tails



(a) $V_{dd} = 0.3V$



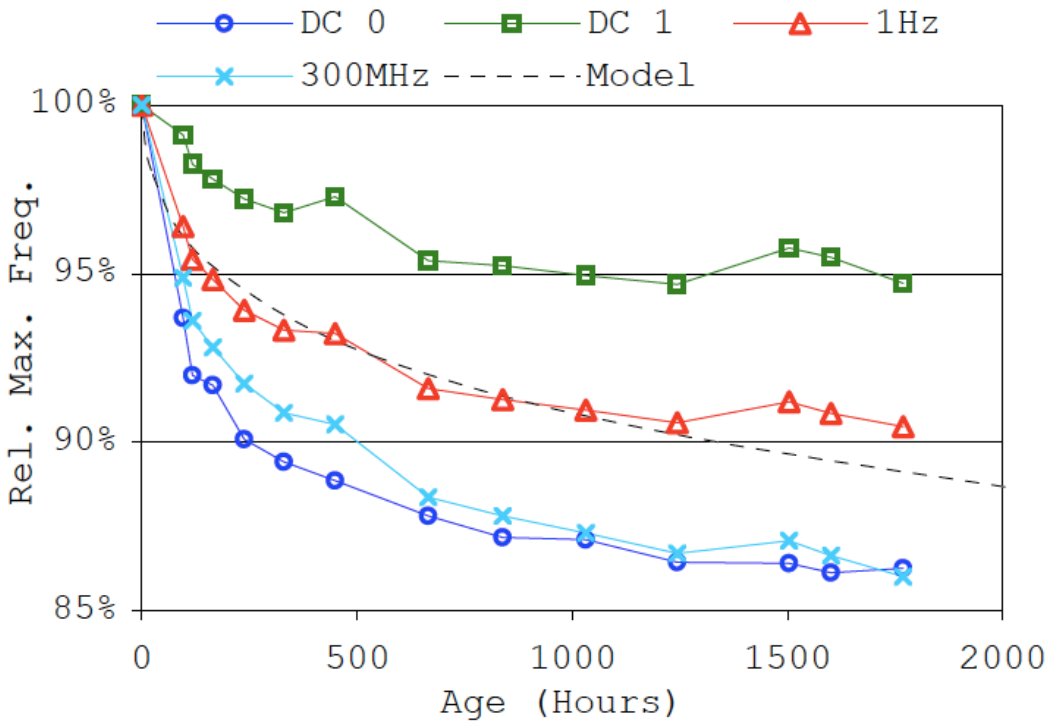
(b) $V_{dd} = 0.8V$

Mehta 2012: 22nm PTM LP – 10,000 samples

Wear Out

- Reduced burnin opportunity
- **Aging:** NTBI, HotCarrier, Electromigration, …
- Forces margin for End-of-Life
  - …and that may not be enough

Cyclone III
(65nm)



[Stott, FPGA 2010]

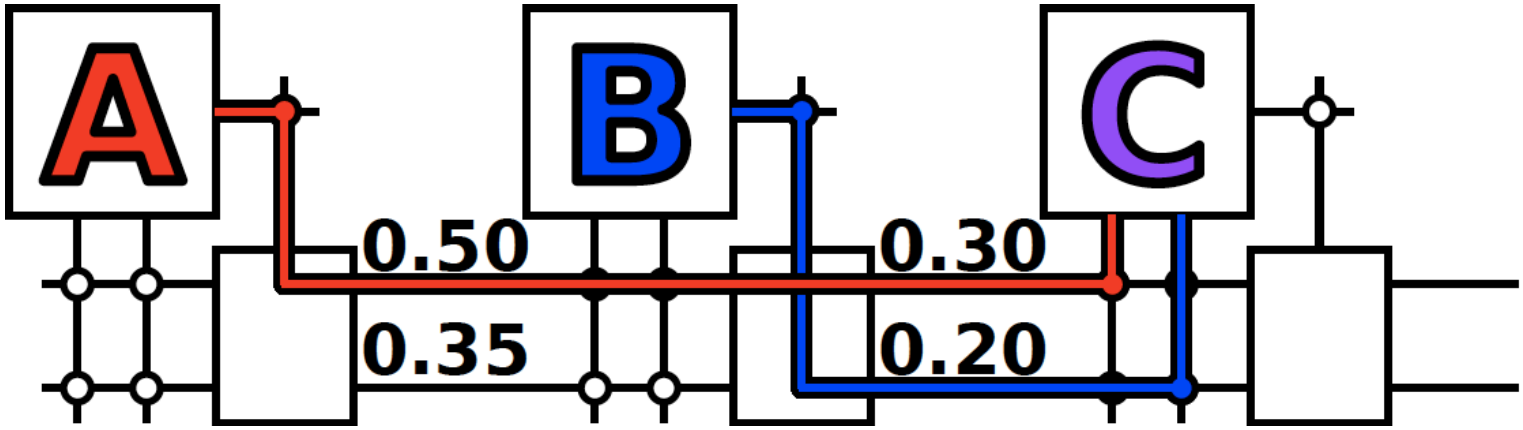
Post Fabrication Configurability

- ASICs bind functions to physical transistors **before** fabrication
  - Before know behavior of device
- FPGAs bind functions to physical devices **after** fabrication
  - After device characteristics determined

Variation Tolerance

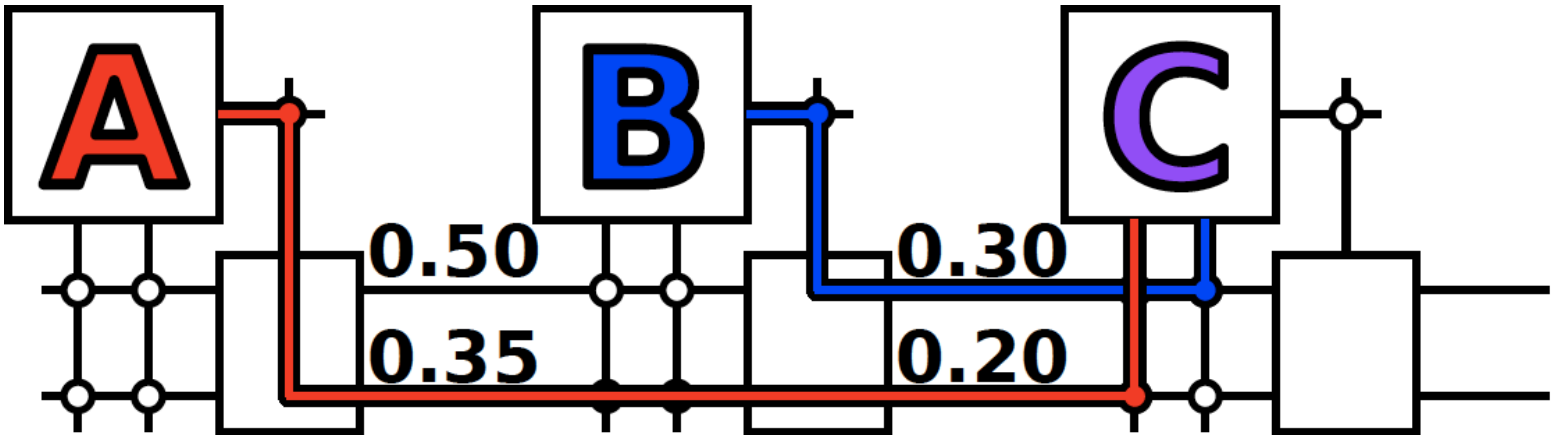
- **Idea:** assign resources, post fabrication to compensate for variations
- **Opportunity:**
  - Balance fast paths and slow paths
  - Assign slow resources to non-critical paths
  - Avoid devices in uncommon tails
  - Scale voltage down more aggressively
- Fixed design limited to worst-case path
  - Must scale voltage up so path meets timing
- **Paradigm shift:** Component-specific mapping

Variation Challenge



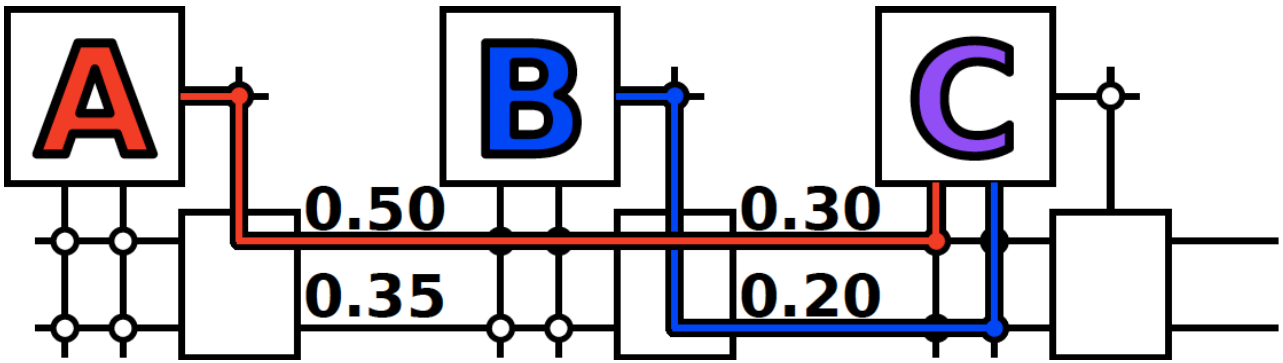
- Use of high V_{th} resource forces high supply voltage (V_{dd}) to meet timing requirement
- Delay: CV/I and I goes as $(V_{dd}-V_{th})^2$

Component-Specific

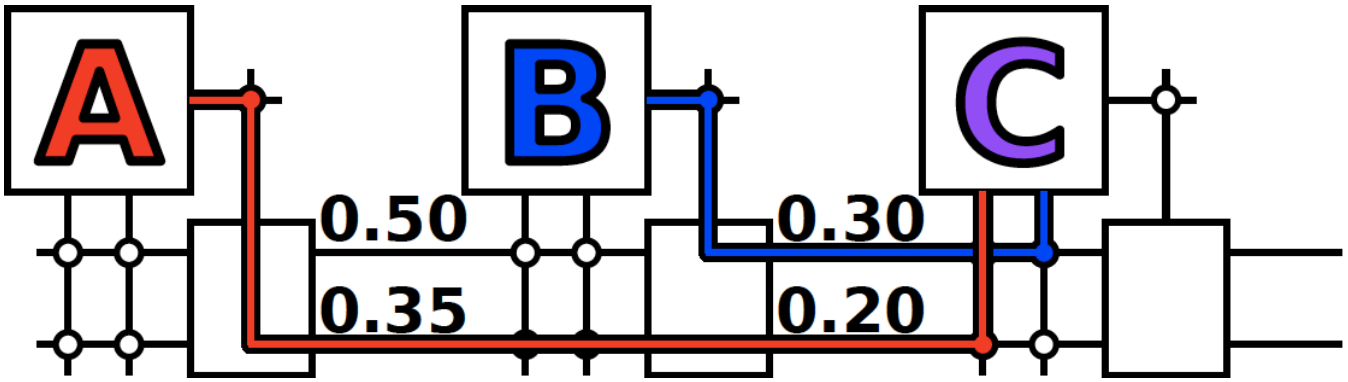


- Avoid high V_{th} resource
- Allow lower supply voltage (V_{dd}) to meet timing requirement
- Delay: CV/I and I goes as $(V_{dd}-V_{th})^2$

Component-Specific Assignment



- Could come out other way
  - Best mapping unique to component

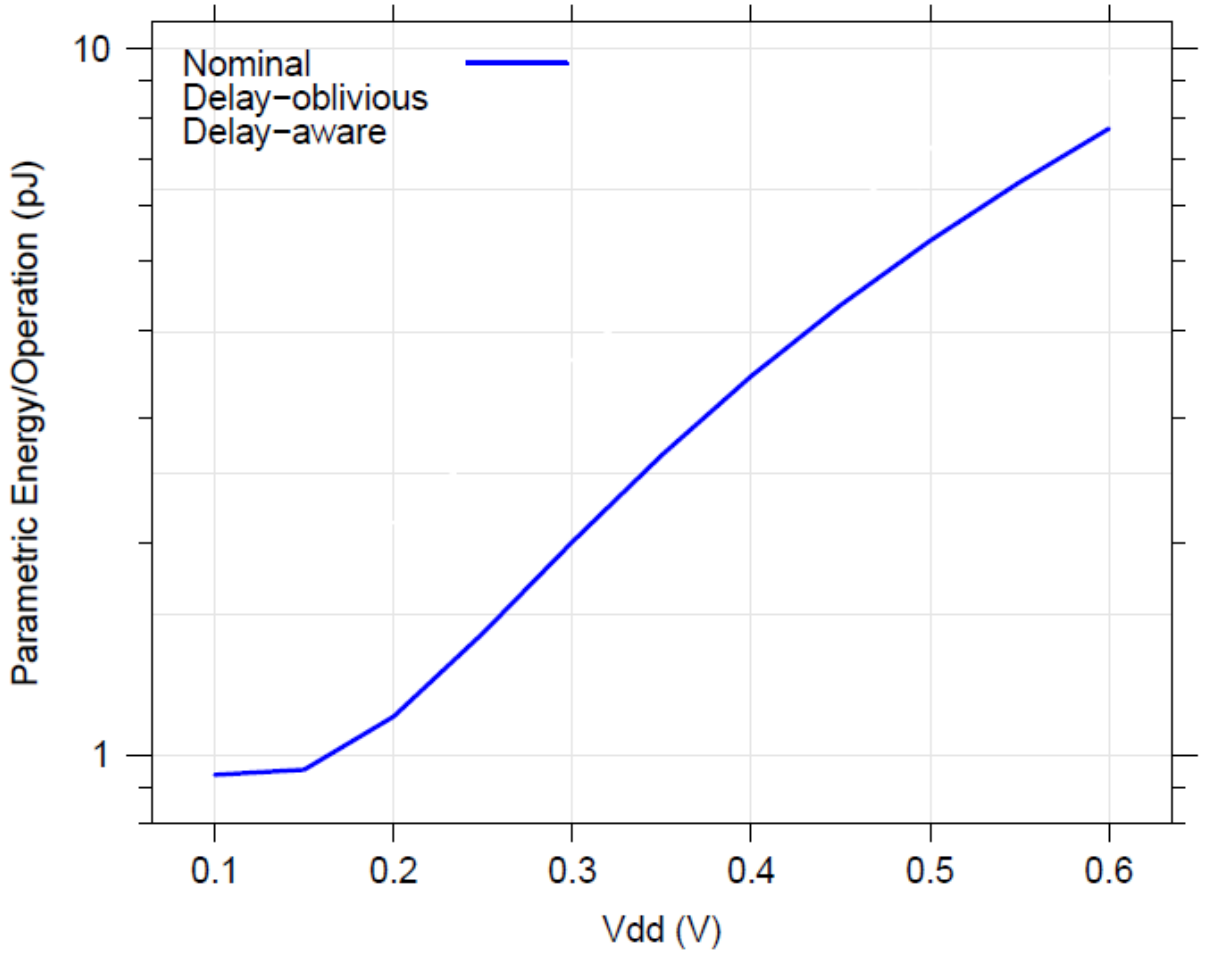


Lifetime Adaptation

- **Issues:** Devices will wear out during operation
  - Parameters vary with time
  - *E.g.* resistance increases → slower operation
- **Opportunity:** reconfigure to replace bad or slow devices
- **Detect:** Concurrent error detection, RAZOR, light-weight application checks, periodic self test
- **Recover:** reassign resources
  - re-invoke mapping
- **Paradigm shift:** mapping throughout lifetime

Energy vs V_{dd} (des)

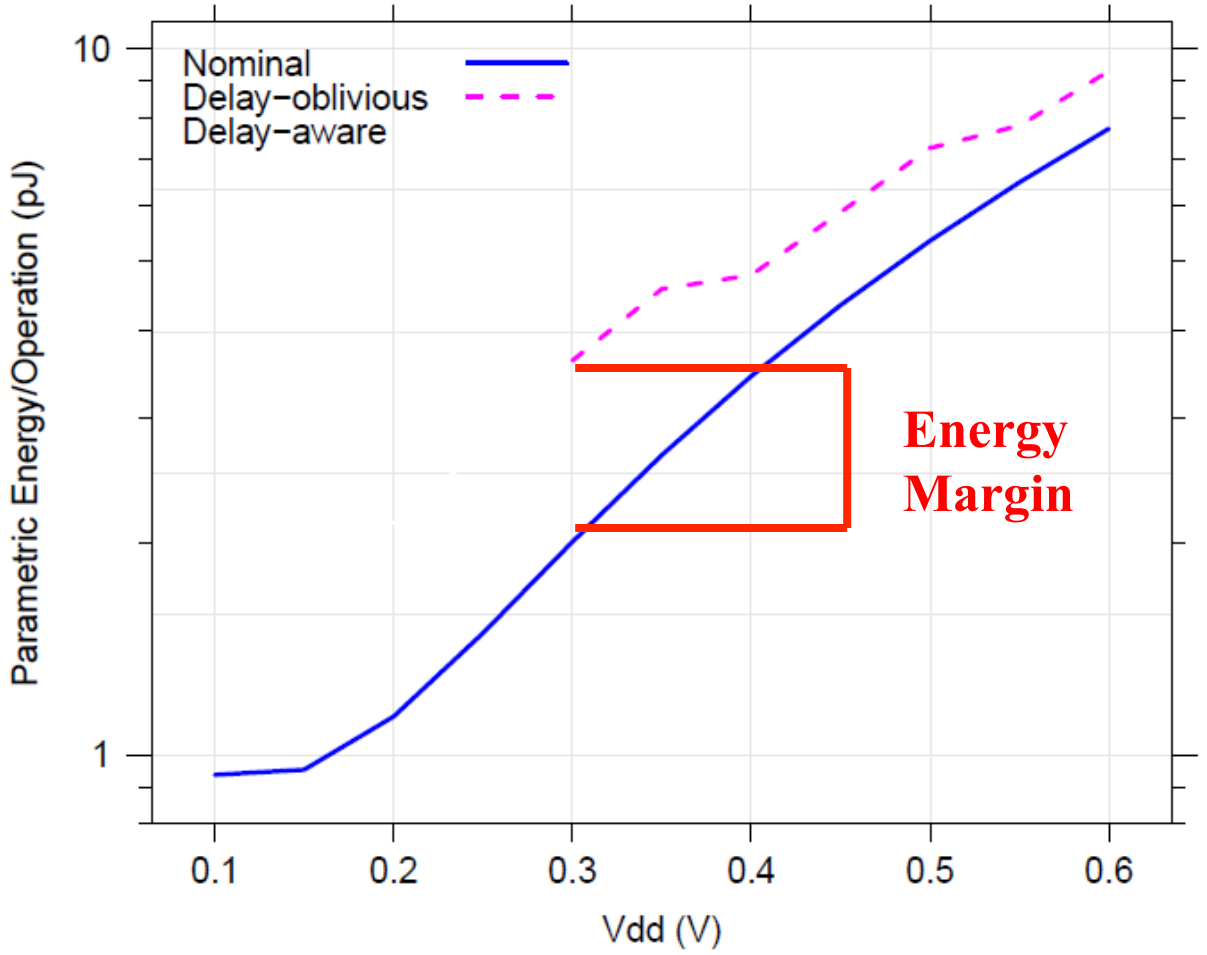
- Nominal uses minimum size



[Mehta, FPGA 2012]

Energy vs V_{dd} (des)

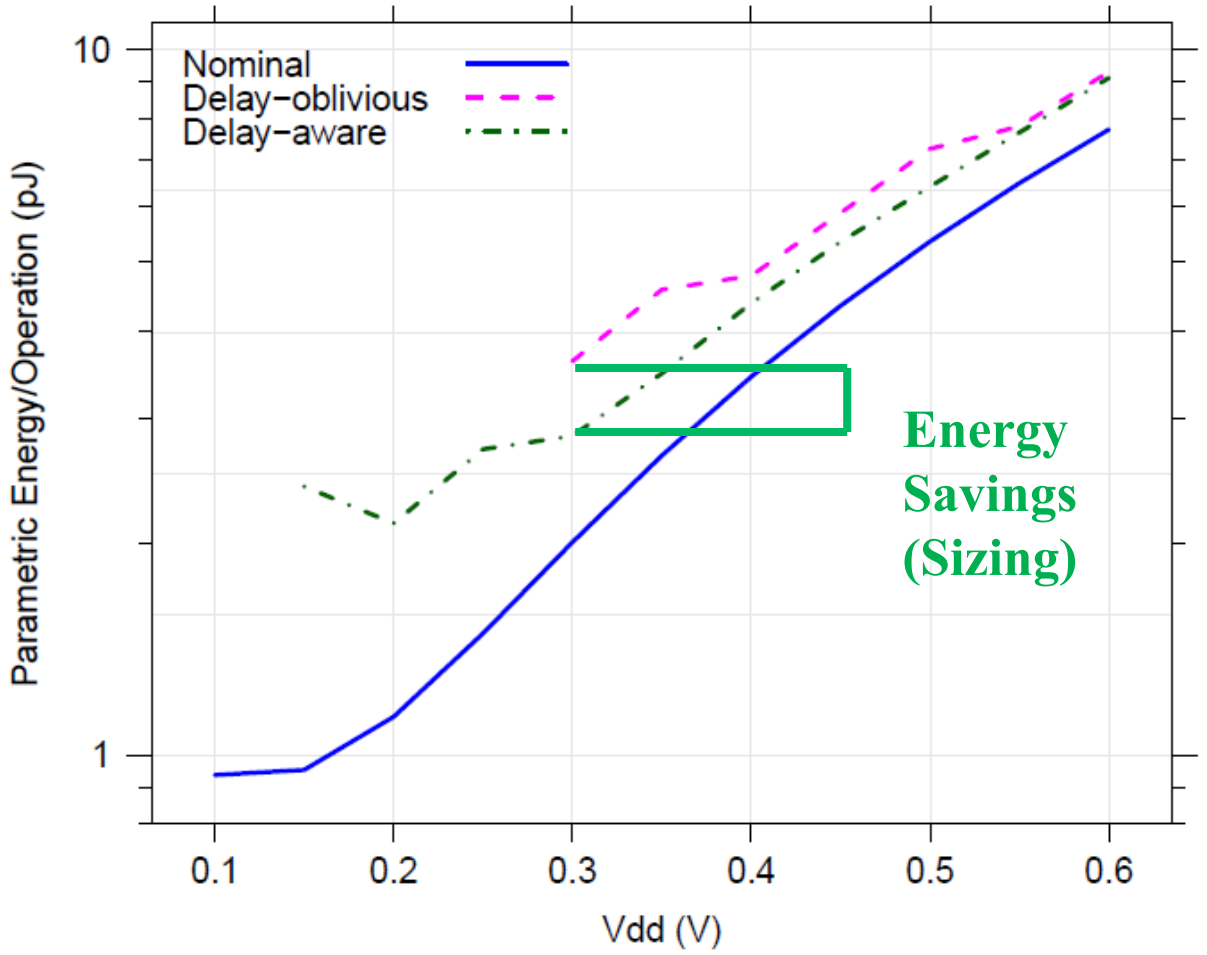
- Nominal uses minimum size



[Mehta, FPGA 2012]

Energy vs V_{dd} (des)

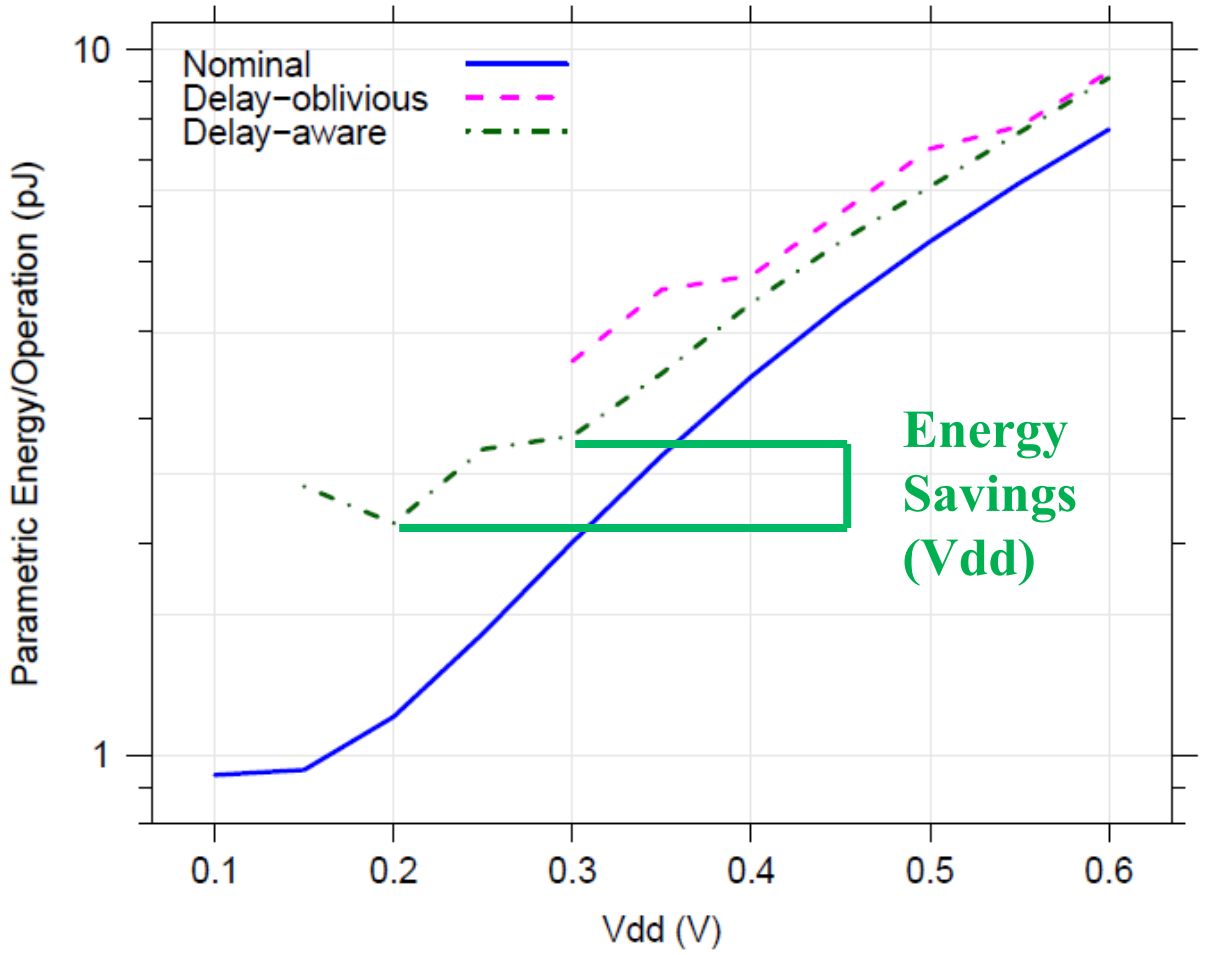
- Nominal uses minimum size
- Delay-aware routing reduces energy margins
  1. Smaller sizes



[Mehta, FPGA 2012]

Energy vs V_{dd} (des)

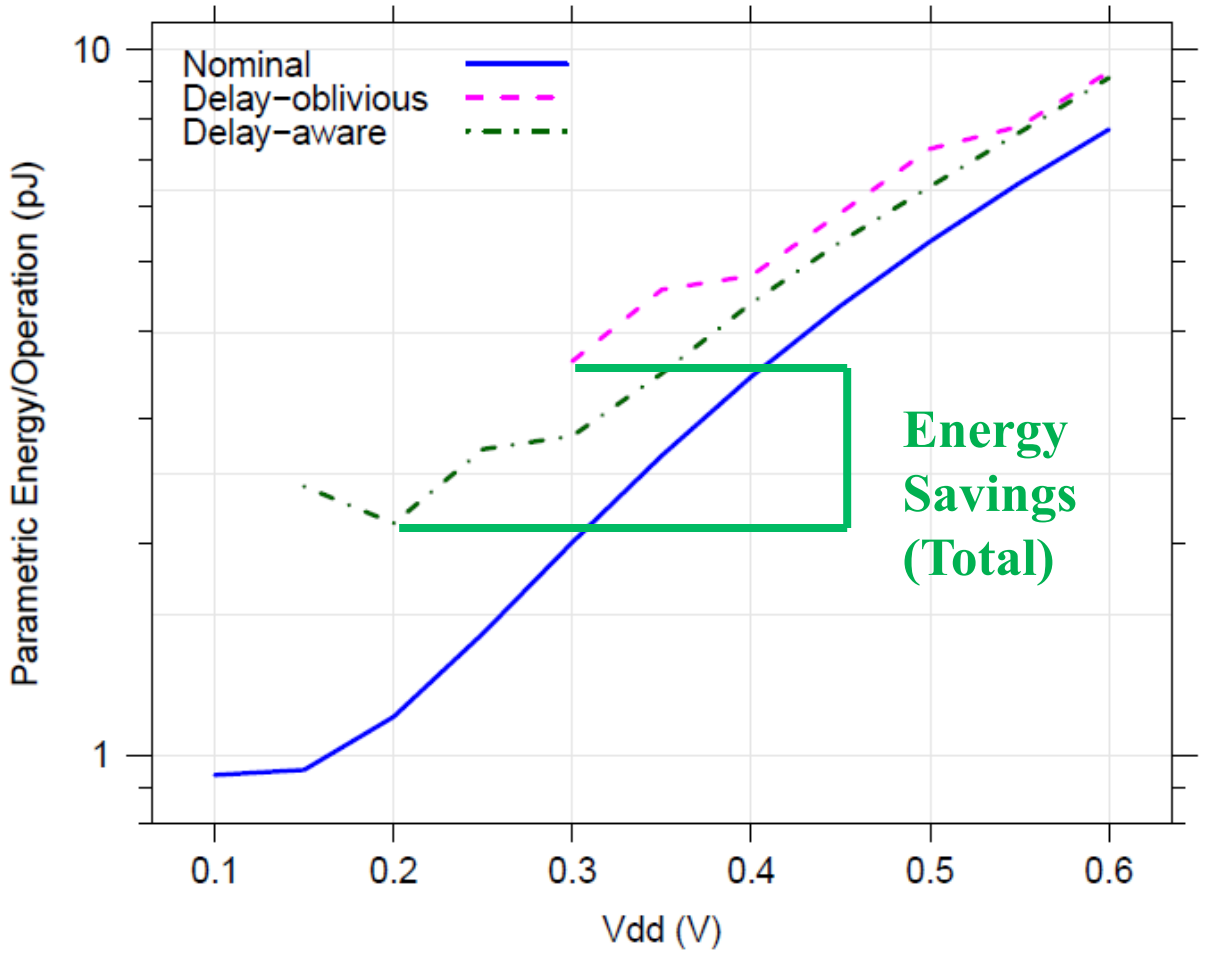
- Nominal uses minimum size
- Delay-aware routing reduces energy margins
  1. Smaller sizes
  2. Lower voltages



[Mehta, FPGA 2012]

Energy vs V_{dd} (des)

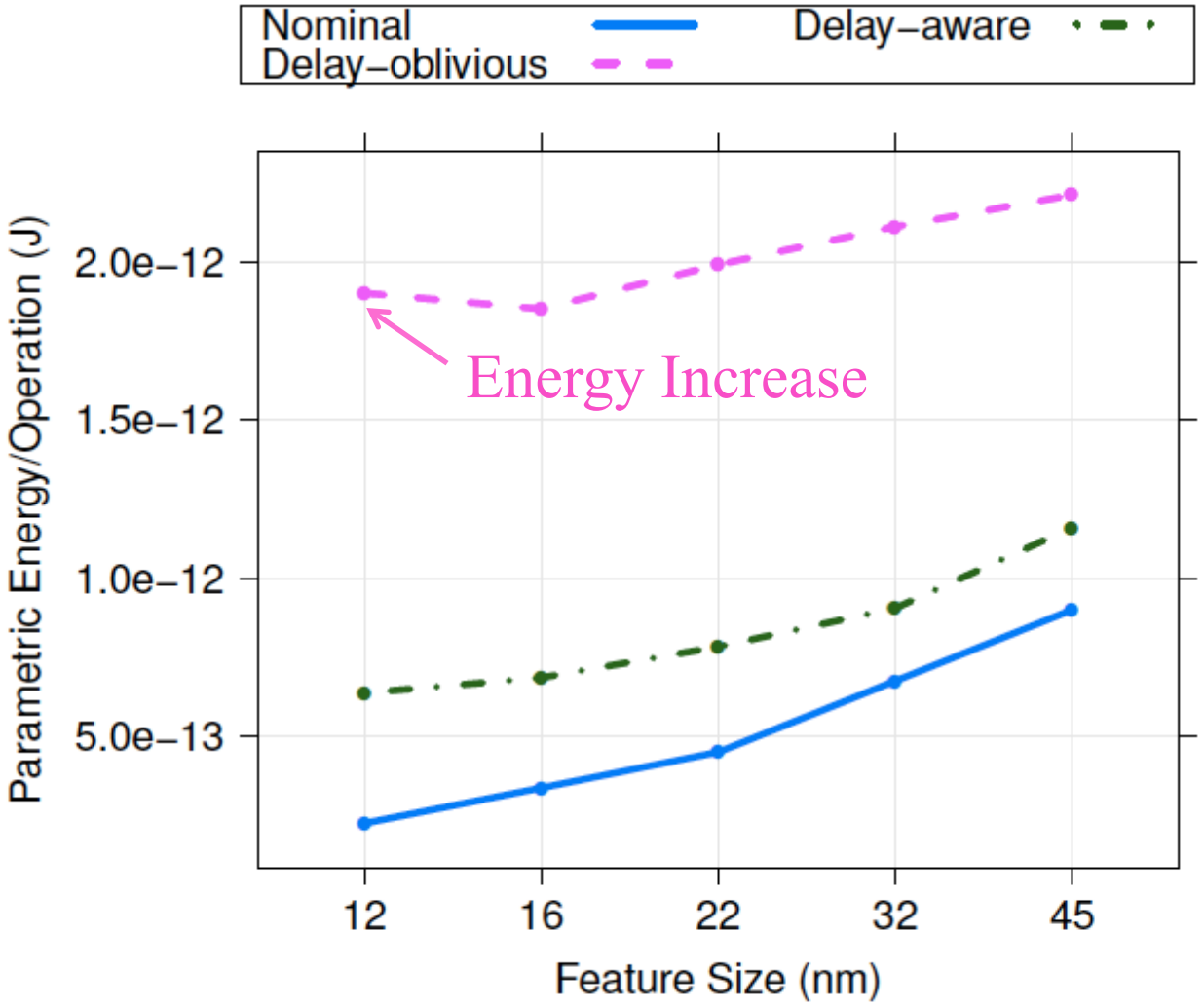
- Nominal uses minimum size
- Delay-aware routing reduces energy margins
  1. Smaller sizes
  2. Lower voltages
  3. Less Leakage



[Mehta, FPGA 2012]

Minimum Energy vs Technology

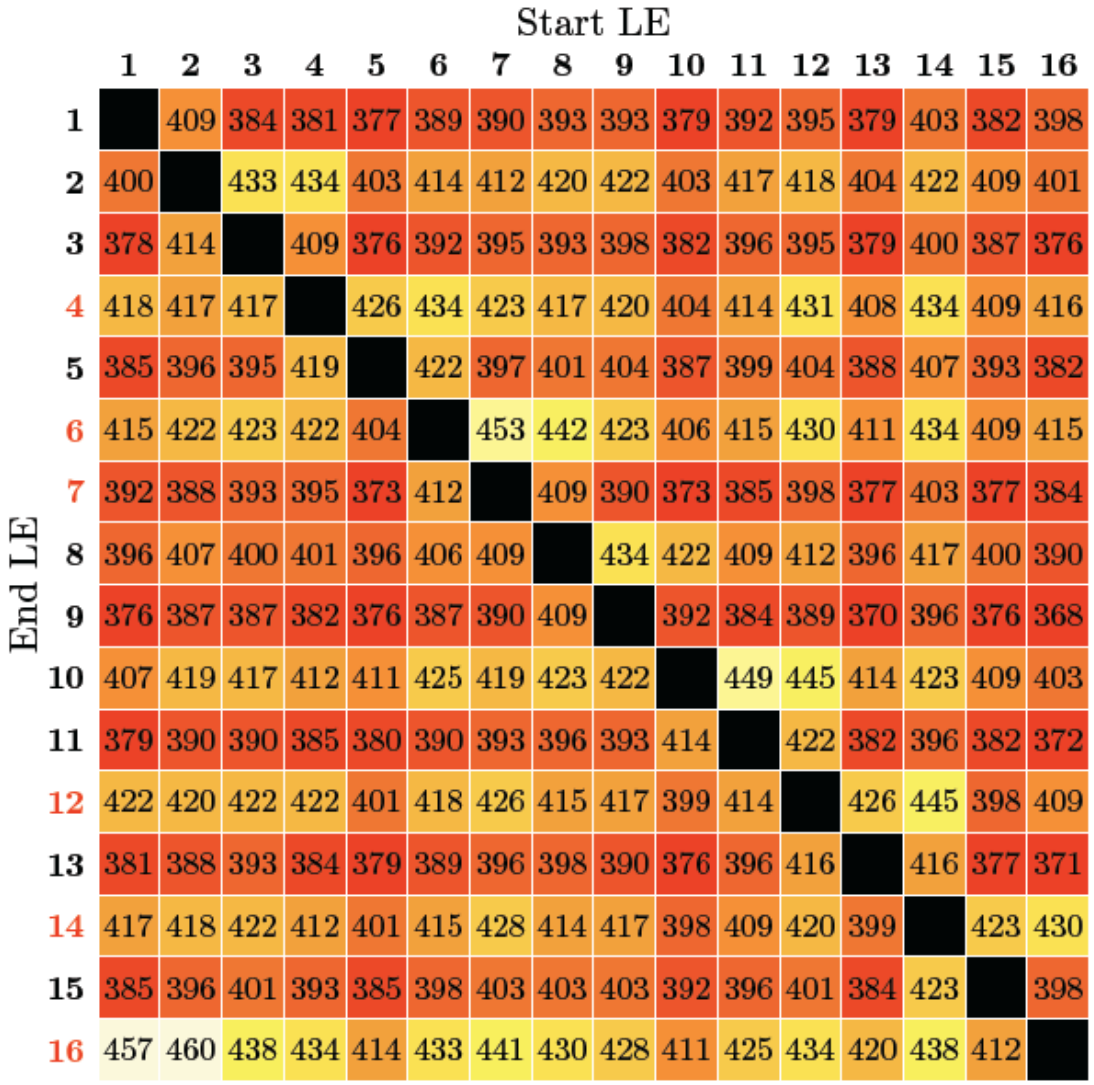
- Delay-oblivious scales to 16nm
- Delay-aware scales to 12nm at least….
- Extend useful life of Silicon technology generation



[Mehta PhD thesis 2012 (refined from FPGA 2012)]

Delay Map

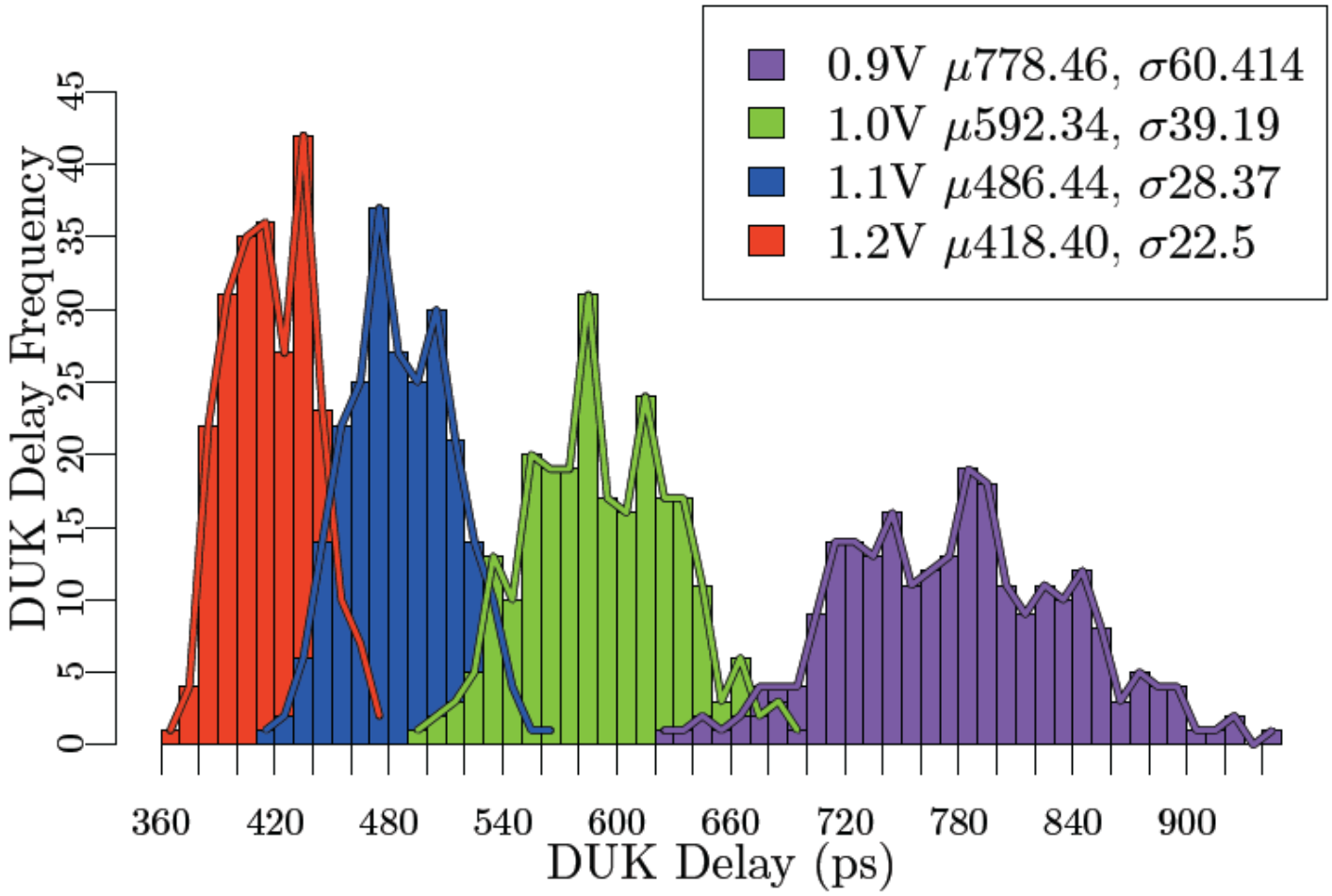
- LAB (27,22)



End LE \ Start LE	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1		409	384	381	377	389	390	393	393	379	392	395	379	403	382	398
2	400		433	434	403	414	412	420	422	403	417	418	404	422	409	401
3	378	414		409	376	392	395	393	398	382	396	395	379	400	387	376
4	418	417	417		426	434	423	417	420	404	414	431	408	434	409	416
5	385	396	395	419		422	397	401	404	387	399	404	388	407	393	382
6	415	422	423	422	404		453	442	423	406	415	430	411	434	409	415
7	392	388	393	395	373	412		409	390	373	385	398	377	403	377	384
8	396	407	400	401	396	406	409		434	422	409	412	396	417	400	390
9	376	387	387	382	376	387	390	409		392	384	389	370	396	376	368
10	407	419	417	412	411	425	419	423	422		449	445	414	423	409	403
11	379	390	390	385	380	390	393	396	393	414		422	382	396	382	372
12	422	420	422	422	401	418	426	415	417	399	414		426	445	398	409
13	381	388	393	384	379	389	396	398	390	376	396	416		416	377	371
14	417	418	422	412	401	415	428	414	417	398	409	420	399		423	430
15	385	396	401	393	385	398	403	403	403	392	396	401	384	423		398
16	457	460	438	434	414	433	441	430	428	411	425	434	420	438	412	

[Gojman, FPGA2013]

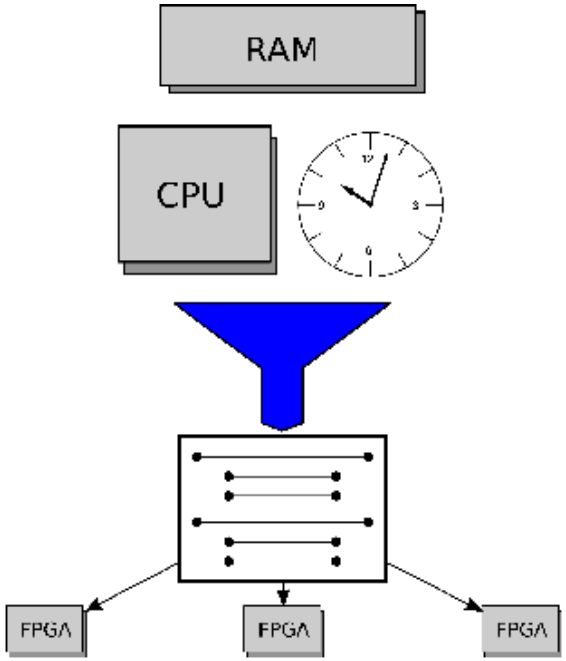
Reduce Vdd (Cyclone IV 60nm LP)

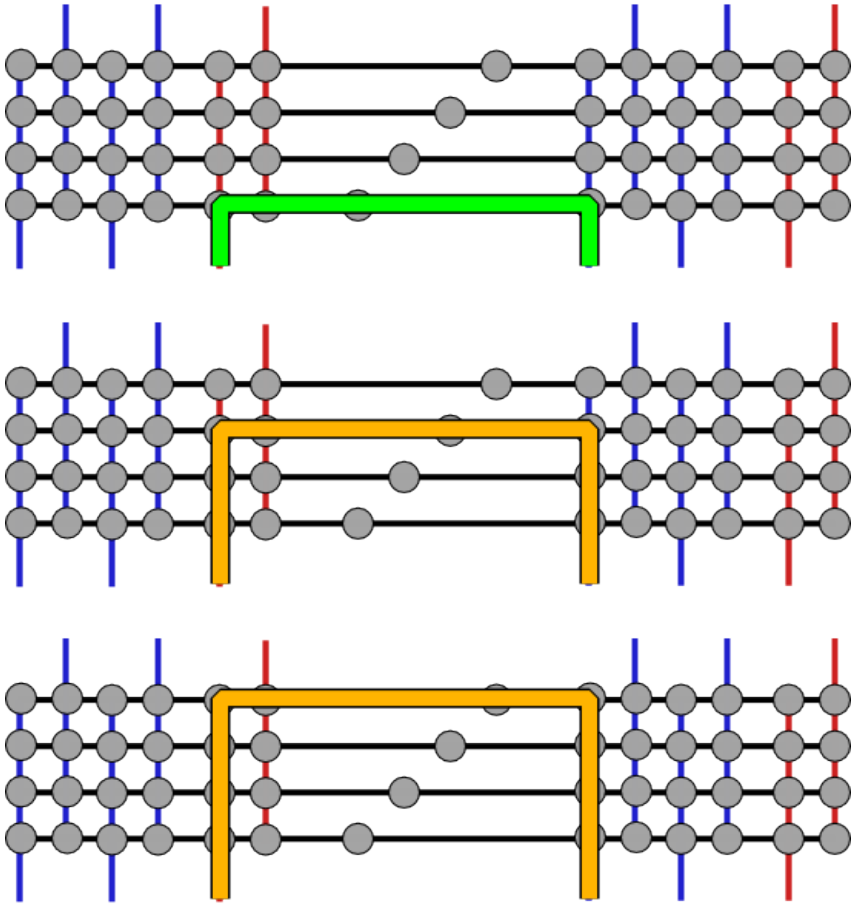


[Gojman, FPGA2013]

Choose **Y**our own **A**dventure

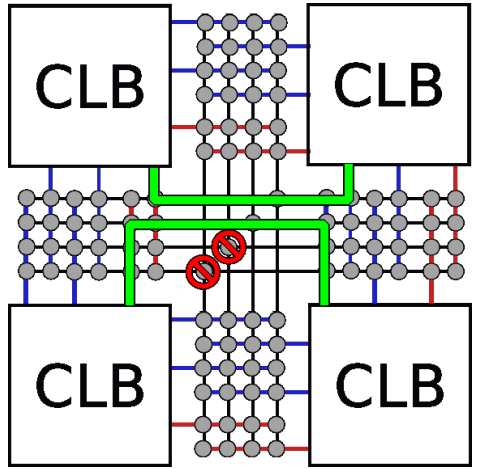
- **Idea:** Precompute Alternate mappings
  - Still just one bitstream



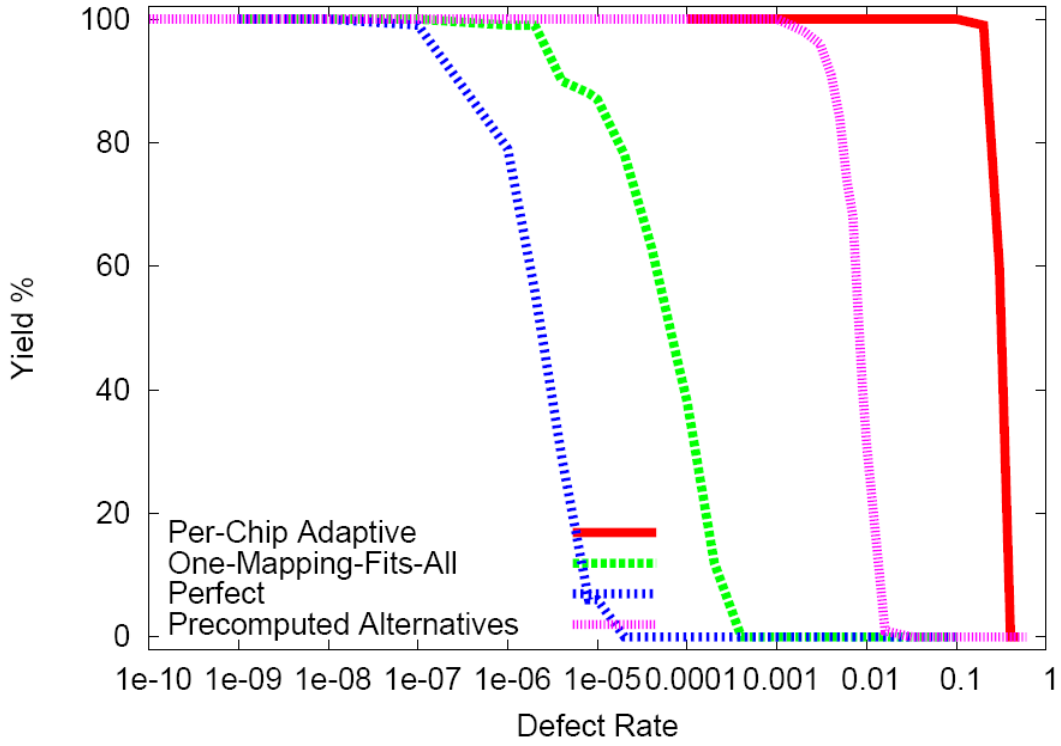


[Rubin, FPGA 2008]

Defect-Level Viable with FPGAs



- Fine-grained repair
- Avoiding routing defects
  - Tolerates >20% switch defects
  - Ongoing work
    - Different defect types and resources
- Don't have to sample tails

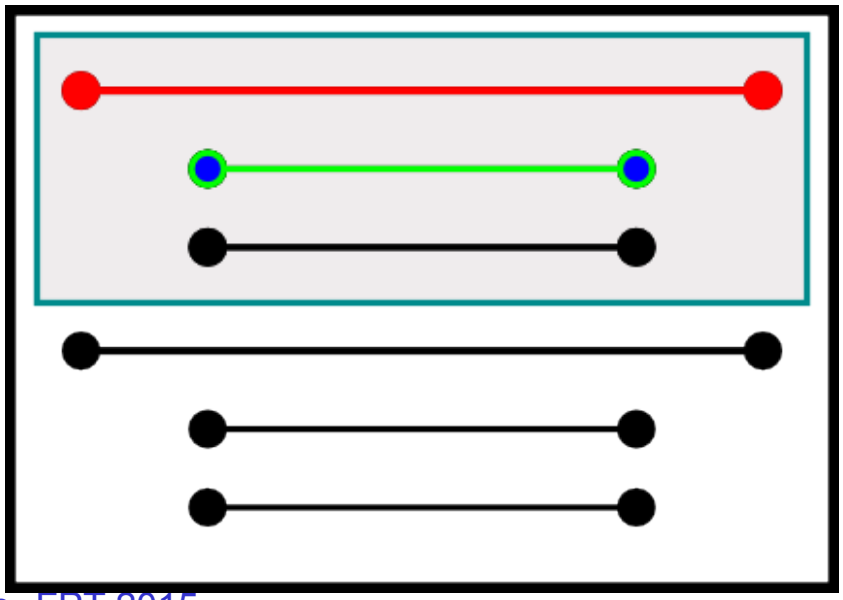


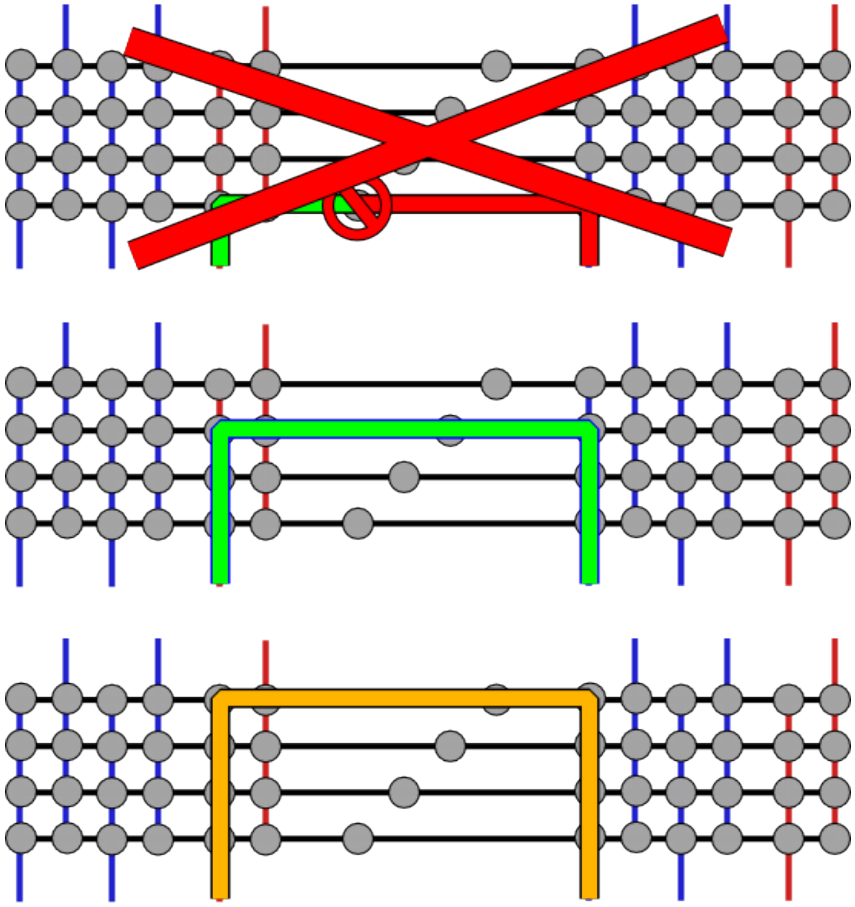
[Rubin/unpublished]

Lifetime Failure?

Go back to alternatives and load again.

- Maybe just the one that failed.
- Millisecond repair.





[Giesen et al./unpublished]

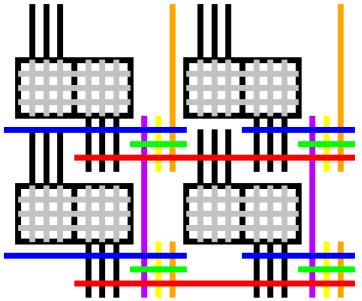
$$E \propto \alpha C V^2$$

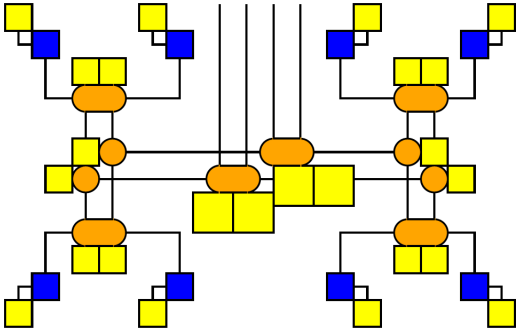
Wrapup

Fund. Energy Benefits of FPGAs

- Architecture
  - Minimize energy in data movement
    - Asymptotic advantage
  - Bound effects of mismatch
- Application
  - Tune parallelism to problem and problem size
  - Specialize to application
- Information – exploit correlation, switching
- Post-fabrication configuration
  - Avoid tails of variation and aging
  - Smaller features, lower voltage

$$E \propto \alpha C V^2$$





ic.ese.upenn.edu

Papers:

- Proc. IEEE 2015: Fundamental Underpinnings
- TRETS 2014:GROKLAB, FCCM2014:GROKINT
- FCCM2014, FPGA2015: Kung Fu, Mem. Arch.
- FPGA 2012: Component-Specific Routing
- TRETS 2011: CYA